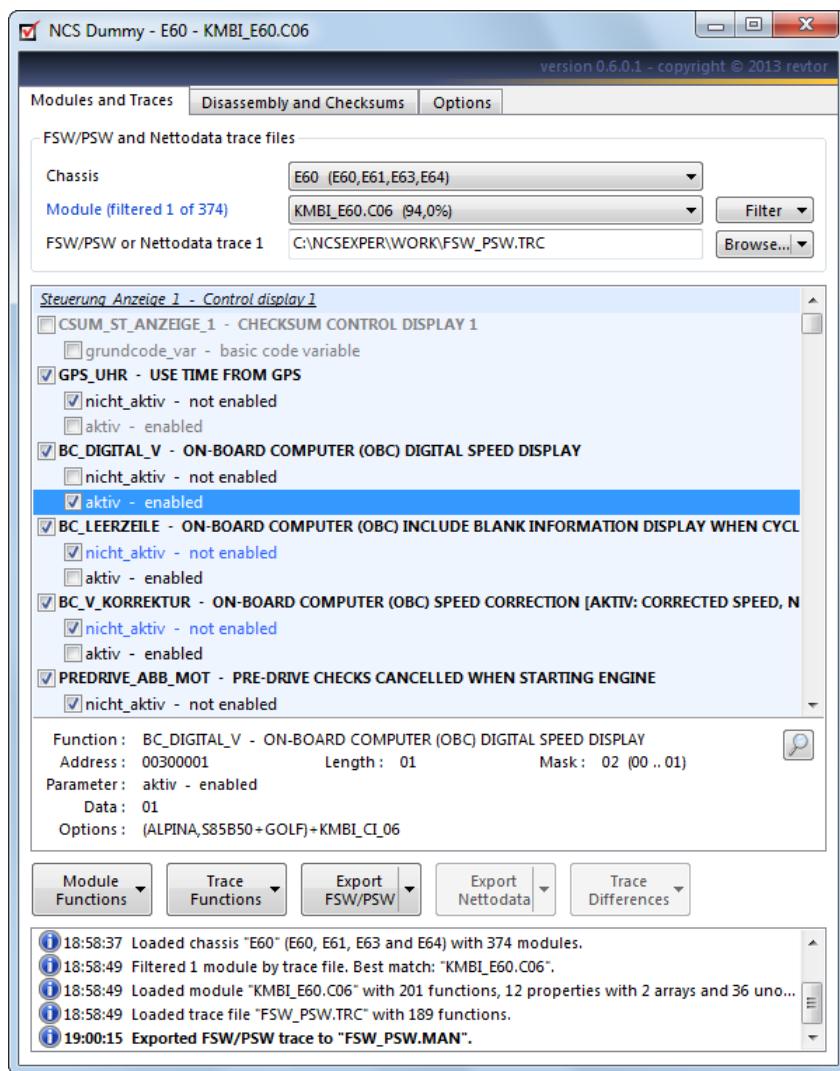


NCS Dummy – Taking the expert out of NCS Expert

NCS Dummy should make life with NCS Expert a bit easier. NCS Dummy parses keyword trace files (`FSW_PSW.TRC`) or nettdata trace files (`NETTODAT.TRC`) and retrieves all possible parameters, control unit data and order options for each function. The parsed trace files can be viewed and modified in a graphical trace editor or exported to text files. NCS Dummy can be used to compare two trace files and show the differences. Furthermore NCS Dummy can disassemble and assemble NCS Expert data files to achieve functionality that is not possible with a stock NCS Expert installation.



This user manual is for **NCS Dummy version 0.6.0.1**. Always check the NCS Dummy thread at [bimmerforums.com](http://forums.bimmerforums.com/forum/showthread.php?t=1553779) (<http://forums.bimmerforums.com/forum/showthread.php?t=1553779>) to make sure you have the latest version of both application and user manual.

If you find bugs or experience any problems with NCS Dummy, please post to the NCS Dummy thread at [bimmerforums.com](http://forums.bimmerforums.com/forum/showthread.php?t=1553779) or contact the author at ncsdummy@revtor.be. When reporting bugs, always include the NCS Dummy log (right click on the log and select "Copy Log" from the menu to put the entire log on the clipboard) and if applicable the keyword trace file (`FSW_PSW.TRC`) or nettdata trace file (`NETTODAT.TRC`) you are trying to parse. Please stay on-topic - NCS Dummy only! Questions concerning where to get or the usage of NCS Expert, what parameters to change to achieve certain functionality on your car,... will not be answered!

1. Table of Contents

1. Table of Contents.....	2
2. Downloading and First Start	3
2.1. Reset Configuration.....	4
2.2. Uninstall.....	4
3. Modules and Traces	5
3.1. Module Functions	5
3.1.1. Translations.....	7
3.1.2. Control Unit Data	8
3.1.3. Order Options	9
3.1.4. Adding New Parameters.....	9
3.2. Trace Functions	11
3.2.1. FSW/PSW	14
3.2.2. Nettodata.....	15
3.2.3. Adding custom data.....	15
3.3. Trace Differences	16
3.4. Searching in the Graphical Trace Editor	19
3.4.1 Shortcut Keys for Search.....	20
3.5. Filter Modules by Keyword.....	20
3.6. Filter Modules by Trace File.....	21
4. Disassembly and Checksums	23
4.1. DATEN files	23
4.1.1. Disassembly and assembly of DATEN files.....	23
4.1.2. Updating checksums in ZST files	26
4.2. NFS files	26
4.2.1. Updating checksums in NFS files.....	26
5. Options	28
6. NCS Expert.....	30
6.1. Reading FSW_PSW.TRC and NETTODAT.TRC trace files.....	31
6.2. Coding modules with FSW_PSW.MAN trace manipulation file	32
6.3. Coding modules with NETTODAT.MAN trace manipulation file	32
6.4. Common NCS Expert errors	33
7. Changelog	34

2. Downloading and First Start

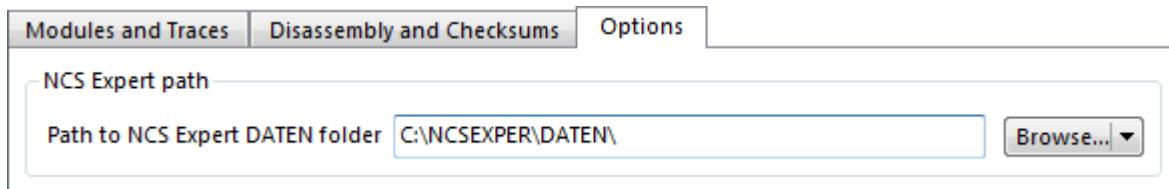
First of all make sure your system meets the minimum requirements for NCS Dummy:

- Microsoft Windows XP or later
- Microsoft .NET Framework 3.5 Service Pack 1
- Minimum display resolution of 800x600 for Windows XP/2003, 1024x768 for Windows Vista/2008/7/8 (at 96 dpi - higher dpi settings require higher resolutions)
- Local or remote access to the NCS Expert "DATEN" folder

Download NCS Dummy (<http://revtor.be/ncsdummy/ncsdummy.zip>) and extract the contents of the downloaded archive to a local folder on your hard disk. Double-click `NcsDummy.exe` to start the application.

If NCS Dummy throws an application error, "*The application failed to initialize properly (0xc0000135)*", you are missing the .NET Framework. Download and install the Microsoft .NET Framework 3.5 Service Pack 1 (<http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=22>). Windows 8 detects the missing framework and will automatically propose to download and install the feature. If your computer is running an older version of the .NET Framework NCS Dummy will ask you to update to version 3.5 service pack 1.

On the first launch NCS Dummy checks for the existence of folder `C:\NCSEXPER\DATEN\`. If the path is not found NCS Dummy automatically switches to the "Options" tab and asks to enter the path to the NCS Expert DATEN folder. After the path has been set NCS Dummy is ready for use.



A lot of information in NCS Dummy can be exported to text files. By default NCS Dummy will open these text files with Microsoft Notepad. If you have a third party plain text editor you prefer, like Notepad++ or UltraEdit, now is a good time to set this in NCS Dummy. Point "Path to custom text file viewer" on the "Options" tab to the text editor's executable and (if necessary) specify the text editor's command line arguments in "Text file viewer arguments" (the default "%1" works for most applications).

While the NCS Dummy graphical trace editor is quite comfortable to use in Windows XP/2003 on a 1024x768 pixels display (at 96 dpi), the larger scaling of Windows Vista/2008/7/8 leaves only a limited amount space available to the graphical trace editor. If you have such a small display and want to squeeze out a few extra rows in the graphical trace editor, launch NCS Dummy with command line argument `/smalldisplay`:

```
NcsDummy.exe /smalldisplay
```

This will slightly decrease the font size (on Windows Vista/2008/7/8) and reduce the log height to maximize the available space for the graphical trace editor. It's not a huge difference but can really make a difference on small displays. If you like the result create a shortcut to the NCS Dummy executable that appends this argument.

2.1. Reset Configuration

NCS Dummy remembers its settings, window size and location between sessions by saving the information to the Windows Registry. You can reset the NCS Dummy configuration by deleting registry key HKEY_CURRENT_USER\Software\revtor\NCS Dummy\ or launching NCS Dummy with command line argument /reset:

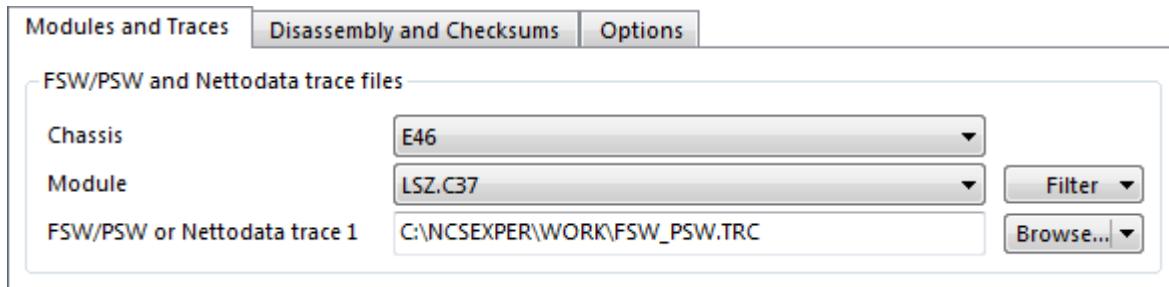
```
NcsDummy.exe /reset
```

2.2. Uninstall

You can uninstall NCS Dummy by removing files NcsDummy.exe, Translations.csv and deleting registry key HKEY_CURRENT_USER\Software\revtor\NCS Dummy\. Alternatively launch NCS Dummy with command line argument /uninstall:

```
NcsDummy.exe /uninstall
```

3. Modules and Traces



There are three main scenarios of use for NCS Dummy Modules and Traces:

- **Module Functions** - Without a trace file loaded the graphical trace editor can be used to view what coding functionality a module offers. In this mode it's even possible to add new parameters to functions and update the NCS Expert data files with these changes.
- **Trace Functions** - Once a trace file is loaded in NCS Dummy, the graphical trace editor shows the currently selected coding options with checkboxes. If you want to make changes to the coding, select the desired options with the checkboxes and then export the modified trace file for coding with NCS Expert.
- **Trace Differences** - When loading two trace files the graphical trace editor shows the differences between both.

Starting from NCS Dummy 0.6.0.0 the graphical trace editor is no longer limited to viewing traces only but because the component is commonly known by that name this manual will keep referring to it as graphical trace editor even when no trace is displayed.

3.1. Module Functions

After selecting a chassis and module NCS Dummy is in the module functions mode. In this mode you can view the coding functionality of a module.

LENKSEITE_LSZ - LIGHT SWITCH (LSZ) DIRECTION

- wert_01 - value 01
- wert_02 - value 02

LAMPENUEBERWACHUNG - Light monitoring

KALTUEBERWACHUNG_BL_MI - COLD MONITORING OF THIRD BRAKE LIGHT

- nicht_aktiv - not enabled
- aktiv - enabled

KALTUEBERWACHUNG_SL_RE - COLD MONITORING OF RIGHT STANDING LIGHTS

- nicht_aktiv - not enabled
- aktiv - enabled

KALTUEBERWACHUNG_RL_3_LI - COLD MONITORING OF LEFT TAIL LIGHT 3

- nicht_aktiv - not enabled
- aktiv - enabled

KALTUEBERWACHUNG_BL - COLD MONITORING OF BRAKE LIGHTS

- nicht_aktiv - not enabled
- aktiv - enabled

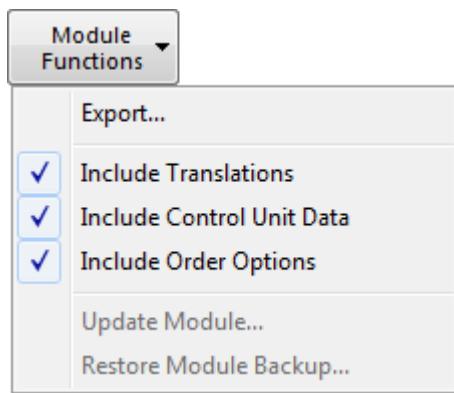
KALTUEBERWACHUNG_BLK_VO - COLD MONITORING OF FRONT TURN SIGNALS

Function :	KALTUEBERWACHUNG_BL - COLD MONITORING OF BRAKE LIGHTS		
Address :	00000100	Length : 01	
Parameter :	nicht_aktiv - not enabled		
Data :	00		
Options :	(CABR,COUP)+(MAERZ03,LED_HECK+(PU01,LSZ_CI_31))		

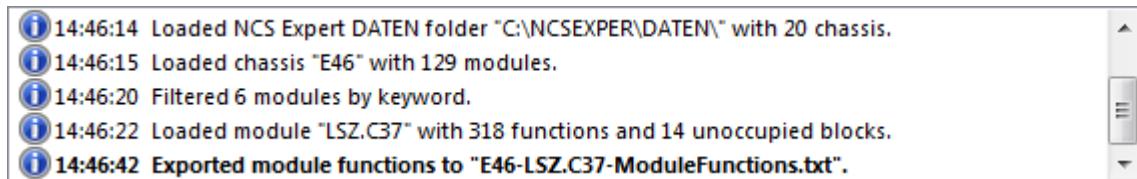
Module functions legend:

- **Group** - Groups usually combine functions and properties that have related functionality.
- **FUNCTION** and **MULTIWORD FUNCTION** - Functions describe certain programmable functionality in a module and have a list of one or more predefined parameters. Functions often allow more data values than parameters are defined. To use custom data values you can add new parameters, add custom data for netto data coding in trace functions mode, or use disassembly. Multiword functions are functions that share the same keyword over two different addresses. Multiword functions are always linked. Changing the function affects the data on both addresses. You can quickly find the counterpart by choosing "Select Multiword Counterpart" from the right-click popup menu.
- **PROPERTY** - Properties are functions that do not have predefined parameters. New parameters cannot be added to properties. The only way to make changes to the data value is by adding custom data for netto data coding in trace functions mode.
- **<UNBELEGT>** - **<UNOCCUPIED>** - Unoccupied blocks are not really unoccupied, but usually contain data that should not be changed afterwards. Just as with properties changes require adding custom data for netto data coding in trace functions mode.
- parameter - Parameters describe the data values for a function.
- new parameter - New parameters can be added to functions from within the graphical trace editor. Note that after updating the module with the changes the added parameters are no longer distinguished from regular parameters.
- invalid parameter - Parameters with invalid data or keyword. These usually only occur when adding parameters and entering wrong information but there are two modules, KMBI_E38.C12 and ARS.COE, that contain an invalid parameter from the factory.
- **<initwert>** - **<initvalue>** - Fake placeholder keyword for initial data value of an unoccupied block.

It is also possible to export the module functions to a plain text file. To do so, click the "Module Functions" button and choose "Export..." from the dropdown menu. The items to be included in the exported file can be set with "Include Translations", "Include Control Unit Data" and "Include Order Options". The graphical trace editor always shows all information but text files have more limited formatting options and may become too overwhelming with all information included.



A successful export operation is indicated in the NCS Dummy log with a bold line. Double-click the bold line to open the exported file immediately in Microsoft Notepad (or your preferred plain text editor when set on the "Options" tab). Right-click the bold line for more options.



Below is an excerpt of exported module functions with control unit data and translations but no order options.

Generated with NCS Dummy version 0.6.0.1 – copyright © 2013 revtor

Output : Module Functions
Chassis : E46
Module : LSZ.C37

FUNCTION KEYWORD PARAMETER KEYWORD	ADDRESS (LENGTH) DATA	MASK MASKED	FUNCTION KEYWORD TRANSLATION PARAMETER KEYWORD TRANSLATION
LENKSEITE_LSZ wert_01 wert_02	0000001E (01) 00 01	80 00 80	LIGHT SWITCH (LSZ) DIRECTION value 01 value 02
LAMPENUEBERWACHUNG	00000100 (1F)		Light monitoring
KALTUEBERWACHUNG_BL_MI nicht_aktiv aktiv	00000100 (01) 00 01	01 00 01	COLD MONITORING OF THIRD BRAKE LIGHT not enabled enabled
KALTUEBERWACHUNG_SL_RE nicht_aktiv aktiv	00000100 (01) 00 01	02 00 02	COLD MONITORING OF RIGHT STANDING LIGHTS not enabled enabled
KALTUEBERWACHUNG_RL_3_LI nicht_aktiv aktiv	00000100 (01) 00 01	04 00 04	COLD MONITORING OF LEFT TAIL LIGHT 3 not enabled enabled
KALTUEBERWACHUNG_BL nicht_aktiv aktiv	00000100 (01) 00 03	18 00 18	COLD MONITORING OF BRAKE LIGHTS not enabled enabled

3.1.1. Translations

Many keywords have English translations but you may notice a lot of blanks when processing your own modules. The translations are a community effort and far from complete. Another issue is that translations do not always really clarify the meaning. The reason for this is that many keywords are reused in several modules and may have different meanings in each. Therefore the translations can't be too specific. This is really troublesome for parameters like "wert_01" (value 01), "wert_02" (value 02),... For these meaningless parameters the control unit data and order options can be helpful (see next chapters).

If you find errors in the translations or know the meaning of untranslated keywords, please mail your corrections or additions to ncsdummy@revtor.be. Please do not make your changes in the `Translations.csv` file. Mail only your corrections or additions!

3.1.2. Control Unit Data

Modules do not store data with keywords like "aktiv" and "nicht_aktiv" but use binary data instead. The *address* is a 32 bit (4 byte) value representing the location in the module's coding memory where the function or group's data can be found and the *length* indicates how many addresses (bytes) are covered by the function or group. The data length can vary but the smallest unit is 1 byte (8 bits) and the data length is always a multiple of that. Many functions don't even need 8 bits to store their parameter data. For example, for "aktiv" and "nicht_aktiv" two states (1 bit) suffice. This is where the mask comes in. The *mask* indicates what bits in a byte are used for the function and allows a single byte to be shared amongst multiple functions. The graphical trace editor shows the possible values for a given mask between brackets. The *data* is the actual parameter data that can be stored in the address location. The *masked* value (only shown in the exported text files) is the data with the mask applied. The masked data is not needed when performing all manipulations through NCS Dummy but may come in handy when working directly with the module's binary data.

Pay attention: All values for *address*, *length*, *data*, *mask* and *masked* are hexadecimal - not decimal!

Understanding control unit data is very important to achieve certain functionality for which no parameters exist. In the example below there are only three predefined parameters for function "SPEEDLOCK_X_KMH_MIN_COE" but the mask is FF (0 to 255). So there are not 3 but 255 possible values for the function. In this case there appears to be a clear relation between the keyword and the data: "nicht_aktiv" is 00, "3_kmh" is 03 and "8_kmh" is 08. In other words; the data is the speed and there's a good chance other speeds can be set than the predefined parameters. More information about that later in this manual.

FUNCTION KEYWORD PARAMETER KEYWORD	ADDRESS (LENGTH) DATA	MASK MASKED
SPEEDLOCK_X_KMH_MIN_COE nicht_aktiv	00300017 (01) 00	FF 00
8_kmh	08	08
3_kmh	03	03

Sometimes the control unit data can also be helpful understanding meaningless parameter keywords like "wert_01" and "wert_02". In the example below function "MIND_ANZ_ZYKL_TIPP_BLK" has two parameters but the keywords are not descriptive as in last example. However in this case the function keyword reveals what the data represents: "MIND_ANZ_ZYKL_TIPP_BLK" or "Mindestanzahl Zyklen Tipp-Blinken". In English that is "Minimum number of flashes for one-touch turn signals" (don't worry; there are translations for this one in NCS Dummy). With one-touch turn signals the indicators flash 3 times when shortly tapping the turn signal lever. Looking at the data this learns us that "wert_01" disables the one-touch turn signal functionality and "wert_02" enables it. Mask 1C (0 to 7) also learns us that the function can accept other values than those two. In other words; there's a good chance that you can change the amount of flashes for the one-touch turn signal functionality.

FUNCTION KEYWORD PARAMETER KEYWORD	ADDRESS (LENGTH) DATA	MASK MASKED
MIND_ANZ_ZYKL_TIPP_BLK wert_01	00340506 (01) 01	1C 04
wert_02	03	0C

3.1.3. Order Options

One of the shortcomings of the translations in NCS Dummy is that they do not always clarify the meaning of functions and parameters. NCS Dummy tries to provide a helping hand by displaying extra information for each parameter that has cross references to the order options. While the order options are not exactly always in a human readable format, they could be a great aid for interpreting meaningless parameters like "wert_01", "wert_02",... The displayed order options can be looked up in the <chassis>ZST.000 (ZCS) or <chassis>AT.000 (FA) file in the C:\NCSEXPER\DATEN\<chassis>\ folder. The example below shows the order options for each parameter of function "KL_PREDRIVE_AUSWAHL" from the E60 instrument cluster.

FUNCTION KEYWORD PARAMETER KEYWORD	OPTIONS
KL_PREDRIVE_AUSWAHL	
wert_01	AFS+! (ACC,ACC_STOP_GO)+!S85B50
wert_02	(ACC,ACC_STOP_GO)+!AFS+S85B50
wert_03	AFS+(ACC,ACC_STOP_GO)+!S85B50
wert_04	!AFS+! (ACC,ACC_STOP_GO)+!S85B50
wert_05	S85B50

Neither function keyword nor parameter keywords provide any information about the meaning of this function and its parameters, but the order options help to understand what parameter is to be used in what situation. The order options syntax uses Boolean logic:

- ! (exclamation mark): Boolean **not**
- + (plus): Boolean **and**
- , (comma): Boolean **or**

Take for example the order options for parameter "wert_04": "!AFS+!(ACC,ACC_STOP_GO)+!S85B50". This could be rewritten as:

```
not AFS and not ( ACC or ACC_STOP_GO ) and not S85B50
```

Looking into E60AT.000 reveals the following:

- AFS (order option 217): Active front steering
- ACC (order option 541): Active cruise control
- ACC_STOP_GO (order option 5DF): Active cruise control with stop & go
- S85B50: 5.0-liter V10 (M5)

While the exact meaning of "KL_PREDRIVE_AUSWAHL" and "wert_04" is still unknown, the order options indicate that "wert_04" should be set for E60 that have no active steering, no active cruise control and no M5 V10 engine.

The order options are very useful when retrofitting equipment or enabling functionality that is only present in certain models since they tell you which parameters should be set.

Note that order options are only displayed when "Load order options" on the "Options" tab is checked. Order options have a severe impact on the loading times of large chassis (like E60 and E89). If you think this is unacceptable uncheck "Load order options" but keep in mind you miss out on a lot of information.

3.1.4. Adding New Parameters

Functions often accept more data values than defined by the parameters. Take for example function "WL_DAUER" in module LM2_E6X.C35 which controls the welcome lights duration in seconds. Even though the function spans a whole byte with mask FF (0 to 255) only one parameter is defined: "wert_01" with data 14 (20 seconds).

<UNBELEGT> - <UNOCCUPIED>

<initwert> - <initvalue>

WL_DAUER - WELCOME LIGHTS DURATION [TIME_S=DATA]

wert_01 - value 01

WL_MAX_DAUER - MAXIMUM WELCOME LIGHTS DURATION AFTER MULTIPLE ACTIVATIONS [TIME_S=DATA]

wert_01 - value 01

Function : WL_DAUER - WELCOME LIGHTS DURATION [TIME_S=DATA]	<input type="button" value="🔍"/>	
Address : 00340508	Length : 01	Mask : FF (00 .. FF)
Parameter : wert_01 - value 01		
Data : 14		
Options : E60,E61,E63,E64		

There are several approaches on getting other data values coded into the module, like nettodata coding or disassembly (see later), but starting from NCS Dummy version 0.6.0.0 it is possible to add new parameters from within the graphical trace editor and update the NCS Expert DATEN files. To do so, right-click on a function or function parameter and choose "Add Parameter" from the popup menu. A new parameter is added to the function and you can enter a custom data value and (optional) keyword. By default NCS Dummy gives new parameters keyword "new_parameter" but you can enter anything you like - the keyword does not even have to exist in the keyword file. You can add as many parameters as you like and parameters may share the same data value, but all parameters must have unique names. Once satisfied with the new parameter(s), update the actual NCS Expert DATEN files by choosing "Update Module..." from the "Module Functions" button dropdown menu. If something is wrong with the entered data, for example an empty or duplicate keyword or data value outside the mask, NCS Dummy will highlight the faulty parameter and field(s) in red and prevent you from updating the module until the fault is corrected.

In the example below function "WL_DAUER" was expanded with six new parameters: "10_seconds" (0A), "20_seconds" (14), "30_seconds" (1E), "40_seconds" (28), "50_seconds" (32) and "60_seconds" (3C). Remember that data values are always hexadecimal!

WL_DAUER - WELCOME LIGHTS DURATION [TIME_S=DATA]

wert_01 - value 01

10_seconds

20_seconds

30_seconds

40_seconds

50_seconds

60_seconds

Function : WL_DAUER - WELCOME LIGHTS DURATION [TIME_S=DATA]	<input type="button" value="🔍"/>	
Address : 00340508	Length : 01	Mask : FF (00 .. FF)
Parameter : 40_seconds		
Data : 28		
Options :		

Adding parameters to modules from within the graphical trace editor should provide an idiot proof alternative to disassembly. Note that adding parameters is only possible in module functions mode and it only works for functions, not properties or unoccupied blocks. Added parameters are not written to the NCS Expert DATEN files until you perform "Module Functions" > "Update Module...". For safety reasons NCS Dummy does not permit editing or deleting existing parameters, but you can edit and delete added parameters until updating the module. After updating the module NCS Dummy reloads the module and is no longer able to detect what parameters are original and what are added. In other words; you cannot edit or delete added parameters from within the graphical trace editor once the module has been updated and the added parameters are no longer displayed in a different color. To make changes to existing parameters or more advanced module modifications you will still need to use disassembly.

For your convenience NCS Dummy creates a backup of the files it will modify. Backup files are saved in the same folder and get suffix `.ncsdummy_backup`. No backup is created when a backup already exists for the file. This ensures the backup file is always the original file. You can restore the module backup any time from within NCS Dummy in module functions mode ("Module Functions" > "Restore Module Backup...").

3.2. Trace Functions

A lot of information from the module functions applies to the trace functions as well, so make sure to read chapter [3.1. Module Functions](#) as well.

Trace functions are essentially the same as module functions but with the vehicle's coding overlaid and the ability to change and export the coding. NCS Dummy automatically switches to trace functions mode after setting chassis, module and loading a trace file. If the trace file is invalid NCS Dummy falls back to module functions mode. Trace functions mode can be distinguished easily from module functions mode by the checkboxes in front of functions and parameters.

To load a trace file, click the "Browse..." button for "FSW/PSW or Nettdodata trace 1" or drag and drop the file to the textbox. The "Browse..." button dropdown menu provides shortcuts to quickly load `FSW_PSW.TRC` and `NETTODAT.TRC`. To reload a loaded trace file press **CTRL+ENTER** or **F5** in the textbox or choose "Reload" from the "Browse..." button dropdown menu. You can avoid having to set the chassis and module manually by using file name structure `CHASSIS-MODULE [-optionaltext].TRC`, for example `E38-IKE.C05-FSW_PSW.TRC`. NCS Dummy will automatically set chassis and module from the file name. If you do not know the module a trace file belongs to, NCS Dummy can detect the best match. More information about this in chapter [3.6. Filter Modules by Trace File](#).

KALTUEBERWACHUNG_NSW_R - COLD MONITORING OF RIGHT FRONT FOG LIGHT
 nicht_aktiv - not enabled
 aktiv - enabled

KALTUEBERWACHUNG_BLK_V_L - COLD MONITORING OF LEFT FRONT TURN SIGNAL
 nicht_aktiv - not enabled
 aktiv - enabled

KALTUEBERWACHUNG_BLK_V_R - COLD MONITORING OF RIGHT FRONT TURN SIGNAL
 nicht_aktiv - not enabled
 aktiv - enabled

KALTUEBERWACHUNG_BLK_H_L - COLD MONITORING OF LEFT REAR TURN SIGNAL
 nicht_aktiv - not enabled
 aktiv - enabled

KALTUEBERWACHUNG_BLK_H_R - COLD MONITORING OF RIGHT REAR TURN SIGNAL
 nicht_aktiv - not enabled
 aktiv - enabled

KALTUEBERWACHUNG_SML - COLD MONITORING OF SIDE MARKER LIGHTS
 nicht_aktiv - not enabled

Function : KALTUEBERWACHUNG_BLK_H_R - COLD MONITORING OF RIGHT REAR TURN SIGNAL
Address : 00340001 Length : 01 Mask : 08 (00 .. 01)
Parameter : nicht_aktiv - not enabled
Data : 00
Options : (E60,E61)+MAERZ07,PU07,MUE_BELEUCHTUNG

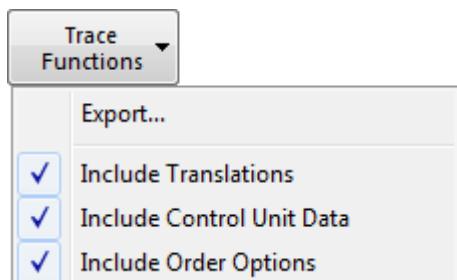
Trace functions legend:

- **Group** - Groups usually combine functions and properties that have related functionality.
- **FUNCTION** and **MULTIWORD FUNCTION** - Functions describe certain programmable functionality in a module and have a list of one or more predefined parameters. Functions often allow more data

values than parameters are defined but you cannot simply add new parameters as in module functions mode. To use custom data values either switch back to module functions mode, add custom data for nettodata coding, or use disassembly. Multiword functions are always linked. Changing the function affects the data on both addresses. You can quickly find the counterpart by choosing "Select Multiword Counterpart" from the right-click popup menu.

- **PROPERTY** - Properties are similar to functions but do not have predefined parameters. The only way to make changes to the data value is by adding custom data for nettodata coding.
- **<UNBELEGT>** - **<UNOCCUPIED>** - Unoccupied blocks are not really unoccupied, but usually contain data that should not be changed afterwards. Just as with properties changes require adding custom data for nettodata coding.
- **UNRESOLVED** and **<UNGELÖST>** - **<UNRESOLVED>** - Unresolved functions are defined in the trace file but do not exist in the module description file. This means almost always the wrong chassis and or module are set for the loaded trace file. By default NCS Dummy uses strict reading and will error on unresolved items but strict reading can be disabled on the "Options" tab.
- **EXCLUDED** - Excluded functions, properties or unoccupied blocks are present in the module description file but not in the trace file. This is perfectly normal behavior. Excluded items can be included by checking one of the parameter checkboxes and included items can be excluded by unchecking the checkbox. Excluded items are ignored by NCS Expert upon coding. Whatever was set before coding will remain after coding.
- **unchecked parameter** - Parameters describe the data values for a function. Gray parameters represent parameters that can be checked but are not checked and were not originally checked.
- **original checked parameter** - Black parameters indicate parameters that are or were originally selected in the loaded trace file. If you made changes to the parameter selection you can always see which parameters were originally checked by the black color.
- **new checked parameter** - Blue parameters are parameters that were originally not checked but now are. There is no functional difference between checked black or blue parameters other than indicating the original checked state.
- **custom checked parameter** - Added custom data for nettodata coding.
- **unresolved parameter** - Unresolved parameters are defined in the trace file but do not exist in the module description file (see UNRESOLVED above).
- **invalid checked parameter** - Parameters with invalid data. These usually only occur when adding custom data and entering wrong information but there are two modules, KMBI_E38.C12 and ARS.C0E, that contain an invalid parameter from the factory.
- **<initwert>** - **<initvalue>** - Fake placeholder keyword for initial data value of an unoccupied block.
- **<tatsächlich>** - **<actual>** - Fake placeholder keyword for parameter data values for which no keyword exists.
- **<benutzerdefinierte>** - **<custom>** - Same as actual, but for manually added custom data.

Trace functions can be exported to a plain text file. To do so, click the "Trace Functions" button and choose "Export..." from the dropdown menu. Similar to the module functions the items to be included in the exported file can be set with "Include Translations", "Include Control Unit Data" and "Include Order Options".



Below is an excerpt of exported trace functions with control unit data and translations but no order options.

Generated with NCS Dummy version 0.6.0.1 – copyright © 2013 revtor

```
Output      : Trace Functions
Chassis     : E60  (E60,E61,E63,E64)
Module      : LM2 E6X.C35
Trace file  : "C:\NCSEXPER\WORK\FSW_PSW.TRC" (FSW/PSW)
```

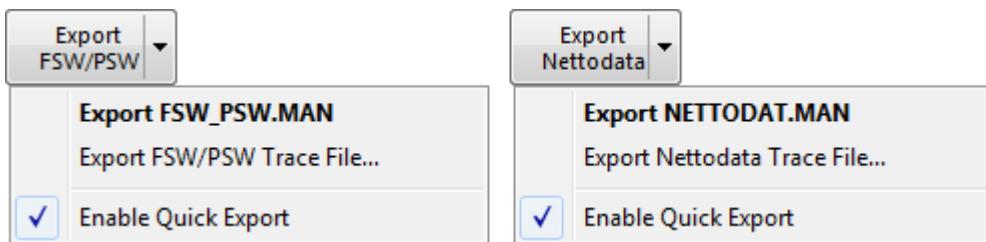
S	FUNCTION KEYWORD PARAMETER KEYWORD	ADDRESS (LENGTH) DATA	MASK MASKED	FUNCTION KEYWORD TRANSLATION PARAMETER KEYWORD TRANSLATION
	KALTUEBERWACHUNG NSW_R	00340000 (01)	80	COLD MONITORING OF RIGHT FRONT FOG LIG
*	nicht_aktiv	00	00	not enabled
*	aktiv	01	80	enabled
	KALTUEBERWACHUNG_BLK_V_L	00340001 (01)	01	COLD MONITORING OF LEFT FRONT TURN SIG
*	nicht_aktiv	00	00	not enabled
*	aktiv	01	01	enabled
	KALTUEBERWACHUNG_BLK_V_R	00340001 (01)	02	COLD MONITORING OF RIGHT FRONT TURN SI
*	nicht_aktiv	00	00	not enabled
*	aktiv	01	02	enabled
	KALTUEBERWACHUNG_BLK_H_L	00340001 (01)	04	COLD MONITORING OF LEFT REAR TURN SIGN
*	nicht_aktiv	00	00	not enabled
*	aktiv	01	04	enabled
	KALTUEBERWACHUNG_BLK_H_R	00340001 (01)	08	COLD MONITORING OF RIGHT REAR TURN SIG
*	nicht_aktiv	00	00	not enabled
*	aktiv	01	08	enabled

Important: Exported trace functions show the trace file coding - not changes you made in the graphical trace editor!

The checkboxes in the graphical trace editor represent the coding state of the module as described by the loaded trace file. The module's coding can be altered by changing the checkbox selections, exporting the changes to a trace manipulation file and recoding the module with NCS Expert and this trace manipulation file. Parameters can be checked but not unchecked. To change the selection of a function simply check the desired parameter. If for whatever reason you do not want a certain function to be included in the exported trace manipulation file, uncheck the function itself. The function is then grayed out and all parameters unchecked. To include an excluded function just check one of its parameters. Excluded functions are ignored by NCS Expert and whatever was set for this function before coding will remain after coding. Therefore it doesn't make much sense to exclude functions but sometimes it can be useful to include a function that is defined in the module description file but not present in the loaded trace file. For each function only one data value can be checked but since multiple parameters may share the same data value it may sometimes appear as if multiple selections are possible. If NCS Dummy displays multiple checked parameters for a function they all have the same data value.

Changes made to a function can be undone quickly by right-clicking the function or its parameters and choosing "Undo Changes" from the popup menu.

Once satisfied with the new coding selection export the modified trace to a trace manipulation file. To do so click the "Export FSW/PSW" or "Export Nettodata" button and choose "Export FSW_PSW.MAN" or "Export NETTODAT.MAN", respectively. If you prefer to save the modified traces under a different name select "Export FSW/PSW Trace File..." or "Export Nettodata Trace File..." instead. When checking "Enable Quick Export" in the dropdown menu the button changes to a split button like the trace file "Browse..." button. Clicking the small right-hand part of the button opens the dropdown menu - clicking the rest of the button immediately exports the trace manipulation file (FSW_PSW.TRC or NETTODAT.TRC depending on the button). Refer to chapters [6.2. Coding modules with FSW_PSW.MAN trace manipulation file](#) and [6.3. Coding modules with NETTODAT.MAN trace manipulation file](#) for more information about coding the changes back to the car.



NCS Dummy supports both FSW/PSW trace files and nettdata trace files. It is important to understand the differences and limitations of both.

3.2.1. FSW/PSW

FSW/PSW stands for function keyword / parameter keyword (Funktionsschlüsselwort / Parameterschlüsselwort in German) and is the primary syntax used by NCS Expert for reading and coding modules. Simply put NCS Expert translates the binary data of a module's coding memory to slightly more readable keywords. When reading a module with NCS Expert, the module's coding is written to trace file `FSW_PSW.TRC`. When coding a module through keyword manipulation with NCS Expert, the coding is read from trace manipulation file `FSW_PSW.MAN`. Below is an excerpt of an FSW/PSW trace file.

```
AUTOM_LICHT_KONTROLLE
    aktiv
HEIMLEUCHTEN
    wert_02
MIND_ANZ_ZYKL_TIPP_BLK
    wert_02
QUITT_BLK_SICHERN
    nicht_aktiv
QUITT_BLK_ENTSICHERN
    nicht_aktiv
TAGFAHRLICHT_ECE
    aktiv
```

The main limitation of this representation is that each function in the module must have a function keyword defined in the module description file and each data value for these functions must have a parameter keyword defined. If no keyword exists the function or parameter cannot be represented in this format. Take for example function "MIND_ANZ_ZYKL_TIPP_BLK". As seen before (chapter [3.1.2. Control Unit Data](#)) this function controls the number of flashes for the one-touch turn signal functionality. The function has 2 predefined parameters - "wert_01" (flash once) and "wert_02" (flash 3 times) - but actually accepts up to flashing 7 times. If you want to recode the module to, say, flashing 5 times that's not possible with the FSW/PSW format because there's no parameter keyword defined for data value 5. Naturally there are ways around that, like adding new parameters in module functions mode, but some changes are simply impossible to achieve through FSW/PSW manipulation. Nettdata (see later) provides an alternative that doesn't suffer from this limitation.

Now assume you managed to code a non-predefined data value to the module for function "MIND_ANZ_ZYKL_TIPP_BLK" and afterwards read an FSW/PSW trace file from the module with NCS Expert (and unmodified DATEN), NCS Expert will not be able to represent the function because there's no predefined keyword for the data value. NCS Expert then omits the function from the FSW/PSW trace file! In the past NCS Dummy didn't display the omitted functions either, but starting from version 0.6.0.0 omitted functions are shown in the graphical trace editor as excluded (unchecked gray functions) and can be included by checking one of the parameters.

Note that with an FSW/PSW trace file loaded, the graphical trace editor only shows items that can be represented in the FSW/PSW format. Properties and unoccupied blocks are not shown. The only way to visualize and manipulate these is through nettdata.

3.2.2. Nettodata

Whenever NCS Expert reads a module's coding to the FSW/PSW trace file, it also creates a nettodata trace file - `NETTODAT.TRC`. This nettodata trace file is actually an intermediate file that is used by NCS Expert when translating the module's binary data to keywords. The nettodata trace file is very interesting because it contains an almost exact binary dump from the module's coding memory. Below is an excerpt of a nettodata trace file.

```
B 00300000,0010,80,07,30,00,00,28,05,AA,FF,00,00,00,00,00,00,00,00,00,00
B 00300100,0010,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
B 00300400,0010,5D,90,42,50,96,1A,23,32,32,01,05,06,0C,43,64,AF
B 00300500,0010,96,1E,40,0F,0A,32,0A,50,78,64,14,32,64,1E,0A,07
B 00300600,0010,0A,00,0A,06,1E,02,3C,96,02,0A,20,07,0D,00,00,00
B 00300700,0010,0A,0D,1E,32,08,0A,32,19,1E,10,40,30,14,3D,19,15
```

When working directly with the module's binary data no keywords are used and the limitations of FSW/PSW traces do not apply. You have total control over the binary data in the module and therefore nettodata manipulation may seem like the holy grail of coding. However, nettodata has serious limitations as well. The biggest issue is the fact that many modules do not support nettodata coding. This is actually rather stupid because modules are obviously not coded with keywords but binary data. Even when coding from an FSW/PSW keyword trace manipulation file, NCS Expert translates all keywords to binary data that is very much like the nettodata format. There's no good reason why a module cannot be coded with nettodata but the job to do so is missing for many modules. If the NCS Expert jobs list does not contain "`NETTODATEN_CODIEREN`", nettodata coding is not supported by the module. From the pre-E60 era not a single module supports nettodata coding...

A second issue is that even when a module supports nettodata coding, it does not always work. Sometimes the process already fails during coding, other times coding is successful but the changes are not working or are lost upon the next restart of the module. It seems NCS Expert performs additional post-processing steps for FSW/PSW coding that are not done for nettodata coding...

These issues make nettodata coding only useful in a few specific cases. In all other cases it's better to use FSW/PSW coding instead and - if custom data values are required - make changes by adding new parameters in module functions mode (preferred) or via disassembly (see later). Despite all this, reading nettodata from modules is still very useful as nettodata trace files often contain more information than FSW/PSW trace files. This extra information can even be useful to achieve certain functionality via FSW/PSW coding.

While you cannot export nettodata trace files when an FSW/PSW trace file is loaded in NCS Dummy (there's not enough information in the FSW/PSW trace file to reconstruct the nettodata), you can export FSW/PSW trace files when a nettodata trace file is loaded. However keep in mind that when a nettodata trace file is loaded, the graphical trace editor shows all functionality of the module and FSW/PSW trace files cannot represent all of it. So making changes to items that cannot be represented by FSW/PSW trace files, exporting the FSW/PSW trace manipulation file and recoding the module with it, obviously will not work. The graphical trace editor displays items that cannot be represented by FSW/PSW trace files in italic.

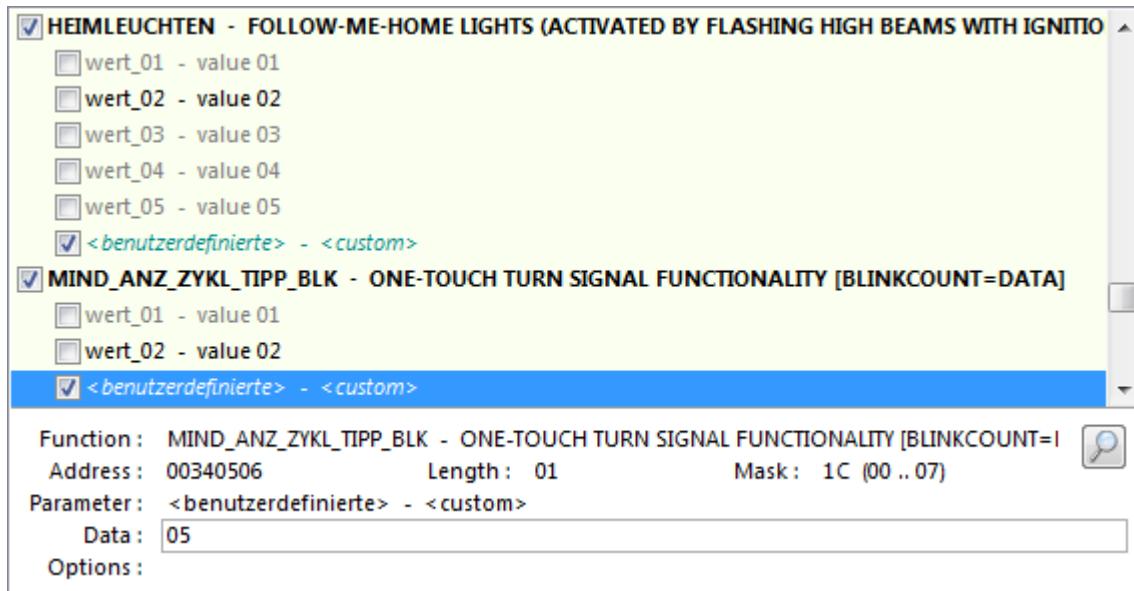
3.2.3. Adding custom data

As seen before (chapter [3.1.2. Control Unit Data](#)) functions often accept more data values than defined by the parameters. Adding new parameters in module functions mode (see chapter [3.1.4. Adding New Parameters](#)) is one way of providing custom data values and works great for functions, but special functions like properties and unoccupied blocks do not support predefined and new parameters. This restriction can be circumvented when working with nettodata where direct manipulation of the binary data is possible without the need of predefined or new parameters.

When a nettodata trace file is loaded NCS Dummy allows you to add custom data for functions, properties and unoccupied blocks. Adding custom data is very similar to adding new parameters in module functions

mode. To do so, right-click on a function or function parameter and choose "Add Custom Data" from the popup menu. A new (fake) parameter is added to the function and you can enter a custom data value for it. Note that unlike with adding new parameters in module functions mode, the NCS Expert DATEN files are not updated and only a single custom data value can be set per function! Once satisfied with the changes to the trace coding, export the NETTODAT.MAN trace manipulation file ("Export Nettdodata" > "Export NETTODAT.MAN") and recode the vehicle as explained in chapter [6.3. Coding modules with NETTODAT.MAN trace manipulation file](#). Do not export to and recode with a FSW/PSW trace manipulation file because FSW/PSW trace files cannot represent the custom data values. If something is wrong with the entered data, for example a data value outside the mask, NCS Dummy will highlight the faulty parameter and data field in red and prevent you from exporting the trace file until the fault is corrected.

In the example below functions "HEIMLEUCHTEN" and "MIND_ANZ_ZYKL_TIPP_BLK" have custom data values applied. Remember that data values are always hexadecimal!

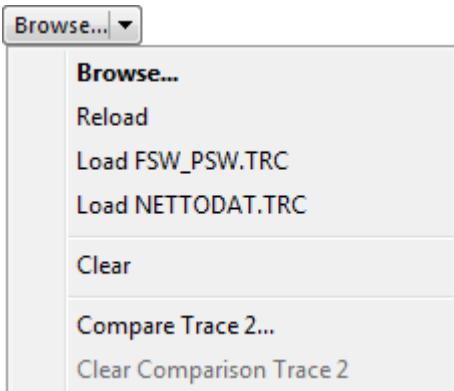


For certain items adding custom data is your only option. However when working with functions (as in the example above) adding new parameters in module functions mode is preferred over adding custom data in trace functions mode. You may still use nettdodata and custom data because it's a quick way to make advanced changes to a module's coding without modifying any NCS Expert DATEN files, but if you experience issues (coding fails or does not stick) use the preferred method instead.

3.3. Trace Differences

The trace differences show the differences between two trace files. NCS Dummy automatically switches to trace differences mode after setting chassis, module and loading two trace files. If the second trace file is invalid NCS Dummy falls back to trace functions mode. Trace differences mode can be distinguished easily from module functions mode and trace functions mode by the black and blue numbered bullets in front of parameters.

Starting from NCS Dummy 0.6.0.0 the second trace file input field is hidden by default to save space for the graphical trace editor. The second trace file can be loaded via the "Browse..." button for "FSW/PSW or Nettdodata trace 1". Choose "Compare Trace 2..." from the button dropdown menu to load the second trace file.



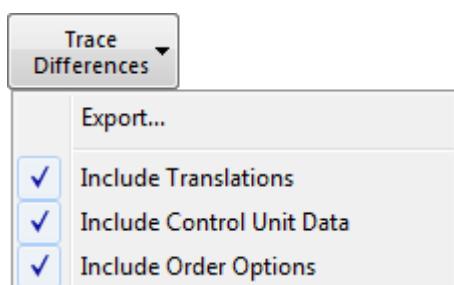
If you use the trace differences functionality often and don't mind losing a few lines in the graphical trace editor you can re-enable the second trace file input field. To do so, uncheck "Hide FSW/PSW or Nettodata trace 2" on the "Options" tab.

Note that you can only load the second trace file when a first one is already loaded for "FSW/PSW or Nettodata trace 1". The trace differences are not a simple text based comparison but compare addresses and binary data - even for FSW/PSW trace files. Therefore you can mix FSW/PSW and nettodata trace files but both must be of the same chassis and module! Trace file 2 is only used for the trace differences. It does not have trace functions.

Trace differences legend:

- **Group** - Groups usually combine functions and properties that have related functionality.
- **FUNCTION** - Functions describe certain programmable functionality in a module and have a list of one or more predefined parameters. Note that in trace differences mode only parameters that are used in both trace files are shown and multiword functions are not highlighted as in module functions mode and trace functions mode (the counterpart may not even be shown if it has no differences).
- **PROPERTY** - Properties are similar to functions but do not have predefined parameters.
- **<UNBELEGT>** - **<UNOCCUPIED>** - Unoccupied blocks are not really unoccupied, but usually contain data that should not be changed afterwards.
- **UNRESOLVED** and **<UNGELÖST>** - **<UNRESOLVED>** - Unresolved functions are defined in the trace file but do not exist in the module description file. This means almost always the wrong chassis and or module are set for the loaded trace file. By default NCS Dummy uses strict reading and will error on unresolved items but strict reading can be disabled on the "Options" tab.
- **TRACE FILE 2** - Functions, properties and unoccupied blocks that occur in the first or both trace files are shown in black. Those that occur only in the second trace file are highlighted in blue.
- **trace 1 parameter** - Black parameters occur in trace file 1
- **trace 2 parameter** - Blue parameters occur in trace file 2.
- **unresolved parameter** - Unresolved parameters are defined in the trace file but do not exist in the module description file (see UNRESOLVED above).
- **<initwert>** - **<initvalue>** - Fake placeholder keyword for initial data value of an unoccupied block.
- **<tatsächlich>** - **<actual>** - Fake placeholder keyword for parameter data values for which no keyword exists.

Trace differences can be exported to a plain text file. To do so, click the "Trace Differences" button and choose "Export..." from the dropdown menu. Similar to the module and trace functions the items to be included in the exported file can be set with "Include Translations", "Include Control Unit Data" and "Include Order Options". Pay attention when one or both trace files are nettodata trace files. You may get unexpected results if you do not include control unit data: If all differences are in the control unit data, the graphical trace editor shows will correctly show the differences but the exported file is empty!



Below is an excerpt of exported trace differences with control unit data and translations but no order options.

```
Generated with NCS Dummy version 0.6.0.1 – copyright © 2013 revtor

Output      : Trace Differences
Chassis     : E60 (E60,E61,E63,E64)
Module      : LM2_E6X.C35
Trace file 1 : "C:\NCSEXPER\WORK\FSW_PSW.TRC" (FSW/PSW)
Trace file 2 : "C:\NCSEXPER\WORK\archive\E60-LM2_E6X.C35-FSW_PSW.TRC" (FSW/PSW)
Differences  : Yes

1 2   FUNCTION KEYWORD      ADDRESS (LENGTH)      MASK      FUNCTION KEYWORD TRANSLATION
          PARAMETER KEYWORD    DATA                MASKED    PARAMETER KEYWORD TRANSLATION
         

WARMUEBERW_RL_BL_1_L      00340006 (01)        01        HOT MONITORING OF LEFT OUTER TAIL LI
```

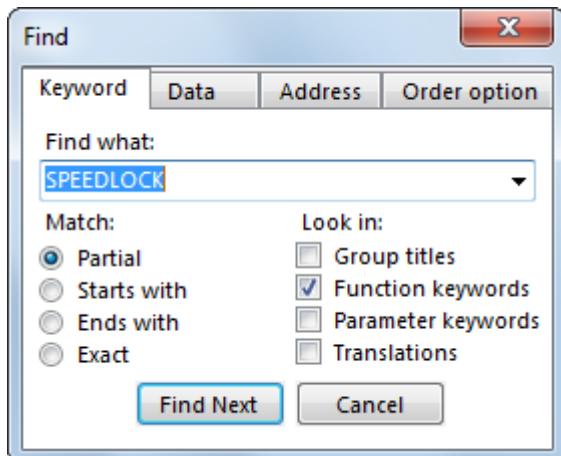
*	aktiv	01	01	enabled
*	nicht aktiv	00	00	not enabled
*	WARMUEBERW_RL_BL_1_R	00340006 (01)	02	HOT MONITORING OF RIGHT OUTER TAIL L
*	aktiv	01	02	enabled
*	nicht_aktiv	00	00	not enabled
*	WARMUEBERWACHUNG_KZL_L	00340006 (01)	10	HOT MONITORING OF LEFT LICENSE PLATE
*	aktiv	01	10	enabled
*	nicht_aktiv	00	00	not enabled
*	WARMUEBERWACHUNG_KZL_R	00340006 (01)	20	HOT MONITORING OF RIGHT LICENSE PLATE
*	aktiv	01	20	enabled
*	nicht_aktiv	00	00	not enabled

3.4. Searching in the Graphical Trace Editor

Some of the larger modules have thousands of lines in the graphical trace editor and finding functions in these just by scrolling is nearly impossible. Therefore NCS Dummy provides powerful search functionality for the graphical trace editor. Open the find window by clicking the magnifier button in the graphical trace editor or press CTRL+F.



The find window offers four search modes; search on "Keyword", "Data", "Address" or "Order option". The search functionality can be used in all modes of the graphical trace editor; module functions, trace functions and trace differences.



The keyword search mode has four match modes: "Partial", "Starts with", "Ends with" and "Exact". In "Partial" match mode, any item that contains the search item will be found (searching on "LOCK" will find "SPEEDLOCK_X_KMH_MIN_COE"). When searching by "Starts with" or "Ends with", only items that have the search item in the respective position will be found (searching on "LOCK" by "Starts with" will not return any results, but searching on "SPEED" will find "SPEEDLOCK_X_KMH_MIN_COE"). The "Exact" match mode will only return items that are identical to the search item ("SPEEDLOCK_X_KMH_MIN_COE" will only be found if that exact text was the search item). Specify the fields to look in when searching with checkboxes "Group titles", "Function keywords", "Parameter keywords" and "Translations" (multiple selections possible). All searches are case insensitive.

The data search mode allows you to search on single bytes or byte arrays in parameter data values. Data bytes are always hexadecimal (0 to FF) and arrays can be entered with each byte separated by a comma, space or no separation at all (for example, "42,4D,57,20", "42 4D 57 20" or "424D5720"). Invalid address or data entries highlight in red and will disable the "Find Next" button. The data search mode provides the same match modes as the keyword search mode; "Partial", "Starts with", "Ends with" and "Exact".

The address search mode has two match modes: "Exact" and "In range". The "Exact" match mode will only return functions that have the same address as the search item. The "In range" search mode will also find

functions that have a different base address, but span over multiple addresses including the search item. Addresses are always hexadecimal (0 to FFFFFFFF).

The order option search mode is only available when "Load order options" is checked on the "Options" tab. It's similar to the keyword search mode but only looks in the order options.

Each search mode remembers the last 10 search items. Use the dropdown to select a previous search item or type the first letters to get a suggestion. Find the next matching item by clicking the "Find Next" button or pressing ENTER or F3. Search upwards by holding SHIFT while searching.

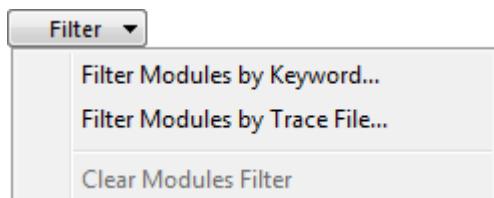
3.4.1 Shortcut Keys for Search

The search functionality for the graphical trace editor provides a number of shortcut keys for quicker access and operation:

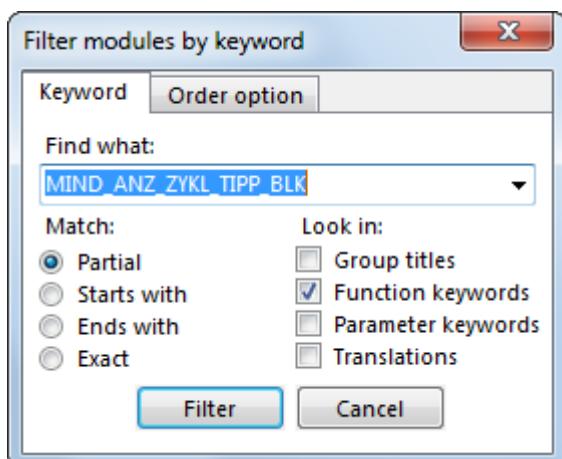
- **CTRL+F** (anywhere on the "Modules and Traces" tab) opens the find window
- **ENTER** (in the graphical trace editor or find window) finds next match
- **F3** (anywhere on the "Modules and Traces" tab) finds next match
- Holding **SHIFT** while searching (ENTER, F3 or clicking the "Find Next" button) searches up
- **ESC** (in the graphical trace editor or find window) closes the find window

3.5. Filter Modules by Keyword

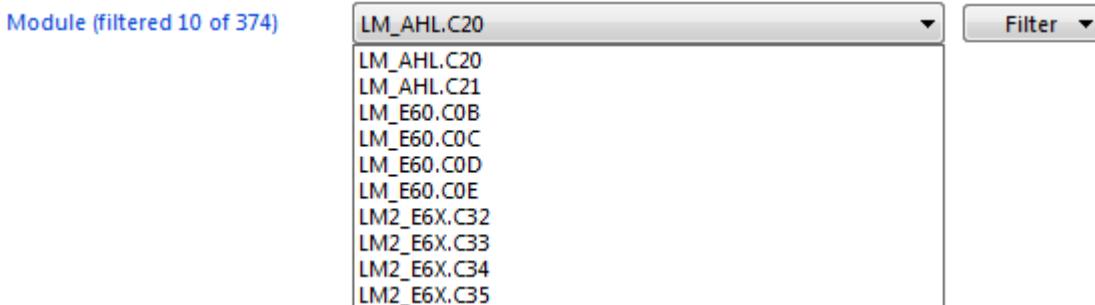
Sometimes you know a keyword or order option but do not know what module(s) it occurs in. Starting from version 0.6.0.0, NCS Dummy allows you to search on keywords and order options in all modules of a chassis. To access the chassis-wide search functionality, set a chassis, click the "Filter" button and choose "Filter Modules by Keyword..." from the button dropdown menu.



The "Filter modules by keyword" window is very similar to the find window for the graphical trace editor (see previous chapter [3.4. Searching in the Graphical Trace Editor](#)), but it lacks the data and address search modes. The search settings and history are shared between the "Filter modules by keyword" and "Find" window.



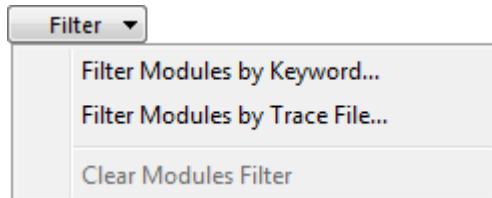
Click the "Filter" button or press ENTER to start the search operation. After the operation completed the "Modules" dropdown is updated and includes only modules that match the search item. The caption is highlighted in blue and shows the number of filtered modules to indicate that the modules filter is active. After filtering the find window is opened automatically with the search item filled in. So if you select one of the filtered modules you can quickly find the occurrence(s) of the search item in the graphical trace editor by clicking the "Find Next" button in the find window or pressing F3.



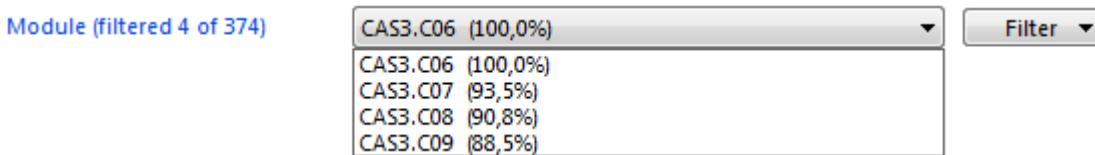
When the modules filter is active you cannot select any modules that are not in the filtered list of modules (this even applies to the automatic detection of chassis and module from the trace file name). To undo the modules filter, click the "Filter" button or right-click the "Modules" dropdown and choose "Clear Modules Filter" from the dropdown menu. It is not necessary to clear the filter first when performing a new filter search operation. If no modules match the search item, no filter is applied and the "Modules" dropdown shows all modules. The modules filter is also cleared automatically when changing or reloading the chassis.

3.6. Filter Modules by Trace File

Sometimes you may have a trace file of which you do not know the module or exact module coding index. Starting from version 0.6.0.0, NCS Dummy can filter the list of modules by an FSW/PSW or nettodata trace file. To do so, set a chassis and load a trace file, then click the "Filter" button and choose "Filter Modules by Trace File..." from the button dropdown menu.



Just as for filtering the list of modules by keyword (see previous chapter [3.5. Filter Modules by Keyword](#)), the caption is highlighted in blue and shows the number of filtered modules when the modules filter is active. Filtered modules are sorted by the match percentage - highest first. The best matching module is selected automatically unless there are multiple modules with equal highest match percentage. If NCS Dummy finds no matching modules at all, the loaded trace file does not belong to the currently set chassis and no filter is applied.

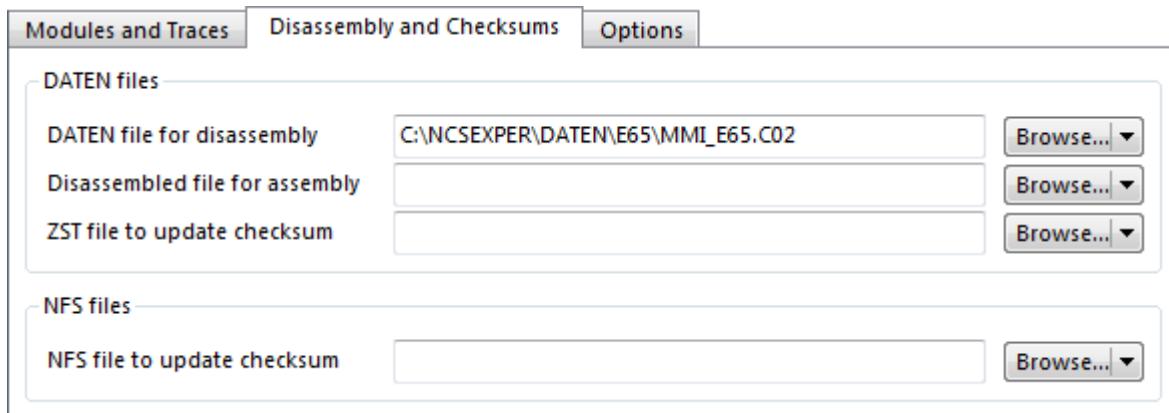


The match percentage serves no other purpose than showing the order of matching. Its absolute value is of no importance and in theory the best match could have a match percentage as low as 50 %. The filtering by trace file is highly accurate and very reliable. If NCS Dummy is able to determine a best match, you can be quite certain that's indeed the correct module for that particular trace file. However, it may occur NCS Dummy does not find a single best match, but multiple best matches (multiple modules that have the same

highest match percentage). If you have both FSW/PSW and nettodata trace files of the module, try the other trace file as well in this case. It's practically impossible you are unable to determine the correct module that way...

Keep in mind that when the modules filter is active you cannot select any modules that are not in the filtered list of modules. To undo the modules filter, click the "Filter" button or right-click the "Modules" dropdown and choose "Clear Modules Filter" from the dropdown menu.

4. Disassembly and Checksums



The "Disassembly and Checksums" tab groups a number of functions that allow manual modification of NCS Expert DATEN files and NFS files.

4.1. DATEN files

DATEN files are the collection of binary and text files in the `C:\NCSEXPER\DATEN\` folder describing the coding functionality of modules and relations to order options for each chassis. These files are used by NCS Expert to read and code modules, and by NCS Dummy to parse the NCS Expert trace files. Modifying these files can be used to achieve functionality that is not possible with a stock NCS Expert installation. Starting from version 0.6.0.0 NCS Dummy is able to perform simple DATEN modifications from within the graphical trace editor (see chapter [3.1.4. Adding New Parameters](#)). This is much easier and more idiot proof than manual disassembly and assembly. Adding new parameters in module functions mode is sufficient for most users in almost every case, but some functionality cannot be achieved from within the graphical trace editor and still requires manual disassembly and assembly of the DATEN files.

4.1.1. Disassembly and assembly of DATEN files

With the disassembly functionality in NCS Dummy you can convert any binary DATEN file to plain text, make changes and then re-assemble to a binary file. This is a very powerful tool but you must understand the purpose of the DATEN files and know exactly what you are doing. If you are thinking about simply copy/pasting functionality from one module to another through disassembly and assembly, you do not understand the concept. It doesn't work like that. The DATEN files describe the coding functionality of modules, but contrary to popular belief they do not affect the modules in any way. Simply put the DATEN files tell NCS Expert (and NCS Dummy) that function X can be found at address Y in the module's coding memory and what settings (parameters) this function accepts. Any functionality that is described in a module's DATEN file must be present in the actual hardware. In other words; you cannot add any functionality to a module through disassembly and assembly that is not already there in the first place. You can only use disassembly and assembly to unlock functions and parameters that are present in the actual module but not described in the DATEN files!

Functions often accept more parameters than defined by the module description file. However it is much easier to add new parameters from within the graphical trace editor than through manual disassembly and assembly.

Gaps in a module's address range may hint at hidden functionality and unoccupied blocks ("UNBELEGT") are not always as unoccupied as the name implies. There is however no reliable way of determining if, where

and what hidden functionality is in a module. While undefined parameters can sometimes be guessed from the function name, order options or other defined parameters, the same cannot be said about undefined functions. Unless you have inside information it is practically impossible to guess where undefined functions are and what they represent...

It goes beyond the scope of this manual to explain what each DATEN file does and how to modify them, but some DATEN files are interesting even without the intention of modifying them. For example, each chassis has a <chassis>CVT.000 coding variants file which describes all keyword coding data for each order option. This is one of the files that is being used by NCS Dummy to generate the cross references to the order options. Below is an excerpt from the E60CVT.000 file disassembled by NCS Dummy.

```
GRUPPE : {"M_Sport"}
AUFTRAGSAUSDRUCK : !(S00F7,S00F8) // !(M5,M6)
FSW_PSW : 101D 0008 // MDRIVE nicht_aktiv
FSW_PSW : 16C1 0008 // ISPEECH MDRIVE nicht aktiv
FSW_PSW : 1393 0009 // MINIMUM_PULSE_FREQ standard
FSW_PSW : 1C88 0008 // CIC_MDRIVE nicht aktiv
AUFTRAGSAUSDRUCK : S00F7,S00F8 // M5,M6
FSW_PSW : 101D 0007 // MDRIVE aktiv
FSW_PSW : 16C1 0007 // ISPEECH_MDRIVE aktiv
FSW_PSW : 1393 047C // MINIMUM_PULSE_FREQ null
FSW_PSW : 1C88 0007 // CIC_MDRIVE aktiv
GRUPPE : {"EDC_M_SPort"}
AUFTRAGSAUSDRUCK : !(S00C5,S00F7,S00F8) // !(EDC,M5,M6)
FSW_PSW : 1601 0008 // EDC nicht_aktiv
AUFTRAGSAUSDRUCK : S00C5,S00F7,S00F8 // EDC,M5,M6
FSW_PSW : 1601 0007 // EDC aktiv
GRUPPE : {"M_Sport_ALBV"}
AUFTRAGSAUSDRUCK : !((S00F7,S00F8)+S00F3) // !((M5,M6)+AKTIVE_LBV)
FSW_PSW : 101E 0008 // MDRIVE_ALBV nicht_aktiv
AUFTRAGSAUSDRUCK : (S00F7,S00F8)+S00F3 // (M5,M6)+AKTIVE_LBV
FSW_PSW : 101E 0007 // MDRIVE_ALBV aktiv
```

It might help to view the disassembled files in a text editor which supports syntax highlighting. C/C++-style syntax highlighting seems to work very well for these files.

All text that is preceded by double slashes (//) are comments added by NCS Dummy. NCS Dummy uses lookups in other files for these comments. The comments give you a quick overview of what functions and parameters are affected by certain order options. That is very useful information for retrofitting equipment. Note that the disassembled <chassis>CVT.000 file does not specify in which module the functions and parameters can be found, but you can use NCS Dummy to locate the affected modules by searching on keywords and order options (see chapter [3.5. Filter Modules by Keyword](#)).

If you want to make changes to a disassembled file, don't make changes to the comments because they are stripped from the file upon re-assembly!

Let's have another look at the example from chapter [3.1.4. Adding New Parameters](#) where function "WL_DAUER" in module LM2_E6X.C35 was expanded from one parameter to seven parameters. Here's how you would achieve the same effect by manual disassembly and assembly. Disassemble module LM2_E6X.C35 and open the disassembled file in a plain text editor. Remember that you can quickly open disassembled files by double-clicking the bold entries in the NCS Dummy log! Locate function "WL_DAUER" in the file. There's only a single parameter defined for the function.

```
PARZUWEISUNG FSW : {00003405} 00000008 0001 14A4 {} (FF) {h} {} // WL DAUER
PARZUWEISUNG_PSW1 : 0026 (14) // wert_01
```

Assume you want to add the same parameters as in the earlier example: "10_seconds" (0A), "20_seconds" (14), "30_seconds" (1E), "40_seconds" (28), "50_seconds" (32) and "60_seconds" (3C). Defining new parameters for the function is as simple as adding new "PARZUWEISUNG_PSW1" lines, but unlike in the graphical trace editor you cannot use parameter keywords. Keyword "wert_01" in the disassembled file is nothing but a comment added by NCS Dummy. The keyword is represented with its identifier 0026 and

between brackets is the data value. In order to be able to use parameter keywords like "10_seconds" we'll have to look up the respective identifiers in the parameter keywords file and if not existing add them.

Disassemble parameter keywords file `SWTPSW05.DAT` and search for "10_seconds", "20_seconds" and so on. As it turns out none of the keyword are already present in the file. Scroll to the end of the file. The last line is a comment added by NCS Dummy.

```
// SWT_EINTRAG : 083A ""
```

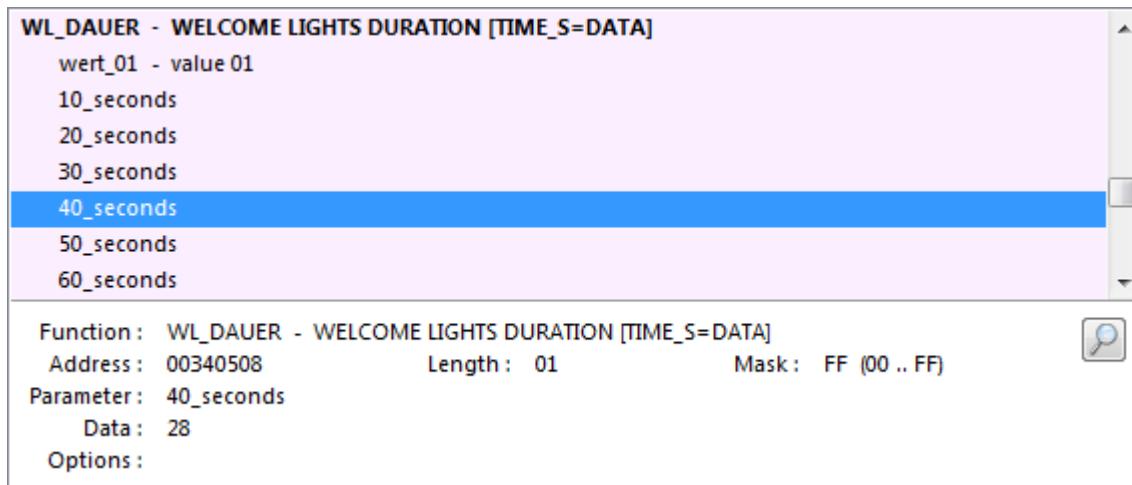
The comment is actually a template for a new entry and its identifier, 083A, is the first highest available identifier. Append the new keywords to the end of the file as follows. Make sure each keyword and identifier are unique!

```
SWT_EINTRAG : 083A "10_seconds"
SWT_EINTRAG : 083B "20_seconds"
SWT_EINTRAG : 083C "30_seconds"
SWT_EINTRAG : 083D "40_seconds"
SWT_EINTRAG : 083E "50_seconds"
SWT_EINTRAG : 083F "60_seconds"
```

Head back to the disassembled `LM2_E6X.C35` module description file and add the new parameters. It may seem tempting to remove the original "wert_01" entry because it is now redundant (parameter 083B shares the same data value) but you should never remove existing parameters as that may break the central coding key (ZCS) or vehicle order options (FA) functionality in NCS Expert! It is safe to have multiple parameters sharing the same value, but you should never add multiple parameters with the same identifier (keyword) and different data values. That will break the module description file!

```
PARZUWEISUNG_FSW : {00003405} 00000008 0001 14A4 {} (FF) {h} {} // WL_DAUER
PARZUWEISUNG_PSW1 : 0026 (14) // wert_01
PARZUWEISUNG_PSW1 : 083A (0A)
PARZUWEISUNG_PSW1 : 083B (14)
PARZUWEISUNG_PSW1 : 083C (1E)
PARZUWEISUNG_PSW1 : 083D (28)
PARZUWEISUNG_PSW1 : 083E (32)
PARZUWEISUNG_PSW1 : 083F (3C)
```

Re-assemble both files and overwrite the original files (make sure you have backups). Now reload the chassis in NCS Dummy or restart the application and you will see the parameters are now available. It should however be clear that adding new parameters from within the graphical trace editor is much easier and less prone to making mistakes.



A few remarks concerning modifying DATEN files... The original function keyword files (`SWTFSW???.DAT`), parameter keyword files (`SWTPSW???.DAT`), and order option keyword files (`SWTASW???.DAT`) are sorted

alphabetically but the sorting order is of no importance. Just add new entries at the end of the file using the template comment added by NCS Dummy. The keywords themselves should have a maximum length of 25 characters and only contain characters A to Z, 0 to 9, % (percentage), & (ampersand), - (minus), . (period), / (slash) and _ (underscore). Function and order option keywords are always uppercase and parameter keywords lowercase.

Functions in module description files (*.C??) are sorted by block, address and mask. In the rare case you have to add functions make sure to respect the sorting order. Parameter order is of no importance though.

4.1.2. Updating checksums in ZST files

In some cases you may want to make changes to the <chassis>ZST.000 and <chassis>ZST.M00 files. Unlike most other DATEN files these files are already in plain text and do not require disassembly. However, modifying the files breaks NCS Expert because they are protected with a checksum at the very end of the file. Use NCS Dummy to update the checksum after making changes to them.

4.2. NFS files

4.2.1. Updating checksums in NFS files

Unlike all other functionality of NCS Dummy, the NFS file checksum updater is not related to NCS Expert but NFS/WinKFP. The *NCS* in NCS Expert stands for *new coding system* whereas *NFS* stands for *new flash system*. Coding is changing the settings (parameters) of a module. Flashing is updating the actual software (firmware) of a module.

It is outside the scope of this manual to explain working with NFS/WinKFP, but for those who want to perform binary firmware modifications without resorting to EEPROM programmers or third party flashing software and interfaces NCS Dummy can provide a helping hand. NFS/WinKFP refuses to flash modified files, but you can circumvent that by using NCS Dummy to update the checksums in the modified files. NCS Dummy supports updating checksums in NFS data files (*.ODA), NFS program files (*.OPA) and NFS NAAB boot update description files (*.OBA). Below is an excerpt of an NFS data file.

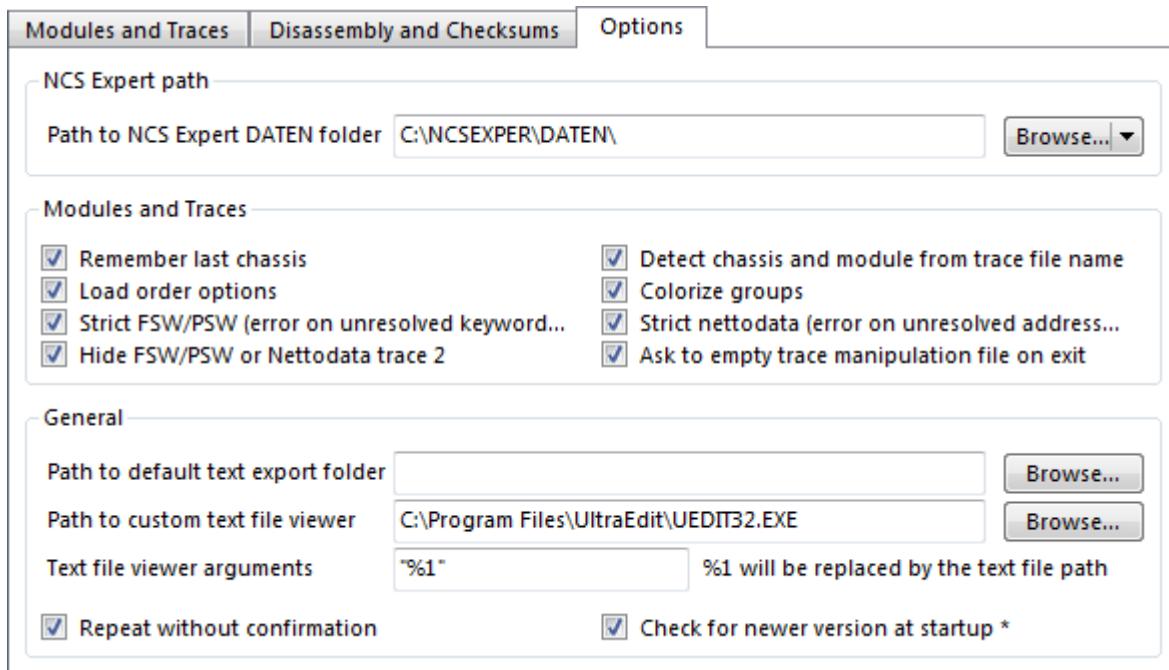
```
;=====
;Austausch-Datei      Daten
=====
$REFERENZ 01118300BE02 2
:020000020000FC
:020000040000FA
:1000000042804E7B08012E4031FC8000FB4021FCE9
:10001000FFFFE000FB4431FC0000FB404EB9000054
:100020004C9A4EB900004D7831FCFFC0FB0431FC06
:10003000000FB00307CC000303C037F429851C878
:10004000FFFC21FCFFFFFFFCAB611FC0001CB5CE7
:100050006000482C48E7C0C0007C070048F81C043A
:10006000C4885238C4B90C380001C4B9662411FCE4
:100070000001C6740C2F0020001067080C2F002709
:100080000010660E11FC0000C67431CFC4BA4FEFE9
:10009000FFE46100662452B8C4986B00132A5378B9
:020000020000FC
:020000040000FA
:004600010AA
$CHECKSUMME 02B2 S
:00000001FF
```

Lines starting with a semicolon (;) are comments and ignored, lines starting with a colon (:) are a derivative of the Intel HEX file format for binary dumps (http://en.wikipedia.org/wiki/Intel_HEX) and lines starting with a dollar sign (\$) are special keywords for NFS/WinKFP. Each Intel HEX line ends with a checksum byte, each special keyword has a single checksum character appended to its value and each NFS file has special keyword \$CHECKSUMME which contains a full-file checksum. NCS Dummy recalculates all of these checksums and (if necessary) asks the user to update the file.

Please note that NCS Dummy only recalculates checksums to make the file valid for WinKFP. NCS Dummy does not interpret the Intel HEX binary data in any way. The firmware in many modules is protected against tampering by an RSA digital signature, so making changes to the binary data will break validation and the module will not work. NCS Dummy cannot fix this! For digitally signed modules you'll have to make sure the binary data in the NFS file is properly signed before making NCS Dummy update the checksums.

5. Options

The default NCS Dummy settings are probably fine for most users, but some may want custom settings. All user configurable settings are grouped on the "Options" tab.



An overview of the available settings is listed below. Changes to settings marked with an asterisk (*) only take effect after restarting the application. All other changes take effect immediately. Settings marked with an exclamation mark (!) should not be changed from the default value unless you know what you are doing.

- **Path to NCS Expert DATEN folder** - The default NCS Expert DATEN folder is `C:\NCSEXPER\DATEN\`, but in a custom NCS Expert installation you may need to specify another path here.
- **Remember last chassis** - NCS Dummy remembers the last set chassis between sessions when this option is checked.
- **Detect chassis and module from trace file name** - When enabled, NCS Dummy tries to automatically set the chassis and module from the file name when loading trace files. In order for this to work the file name must comply to the `CHASSIS-MODULE[-optionaltext].TRC` file name structure, for example `E38-IKE.C05-FSW_PSW.TRC`. You can temporarily disable the automatic detection by holding down SHIFT while clicking "Browse..." or pressing SHIFT+ENTER in the trace file input field.
- **Load order options** - If set, NCS Dummy loads the order options for each chassis and displays extra information for each parameter that has cross references to the order options. Order options have a severe impact on the loading times of large chassis (like E60 and E89). If you think this is unacceptable uncheck this option but keep in mind you miss out on a lot of information.
- **Colorize groups** - Each group gets a different background color in the graphical trace editor when this option is enabled. Disable this option if you prefer a white background.
- **Strict FSW/PSW (error on unresolved keywords) !** and **Strict nettdodata (error on unresolved addresses) !** - When enabled, NCS Dummy throws errors on all unresolved keywords in FSW/PSW trace files and all unresolved addresses in nettdodata trace files. This is good because unresolved keywords and addresses always indicate that you are parsing a trace file with the wrong module. Do not disable these options unless you know exactly what you are doing!
- **Hide FSW/PSW or Nettdodata trace 2** - If checked, the second trace file input field is hidden to save space for the graphical trace editor. A second trace file can be loaded for trace differences via trace 1 "Browse..." > "Compare Trace 2...". If you use the trace differences functionality often and don't

mind losing a few lines in the graphical trace, uncheck this setting to show the second trace file input field.

- **Ask to empty trace manipulation file on exit** - When coding different chassis it is important to empty (not delete) the `FSW_PSW.MAN` trace manipulation file when changing chassis in NCS Expert. Forgetting this may lead to mysterious "COAPI-1141: FSW faulty" or "COAPI-1142: PSW faulty" when reading trace files. If this setting is enabled NCS Dummy asks to empty the trace manipulation file on application exit (if present and not already empty). Disable the setting if you work mostly with the same chassis and find the popup on application exit too obtrusive.
- **Path to default text export folder** - When empty, NCS Dummy sets the default export folder to `C:\NCSEXPER\WORK\` for module function, trace function,... text files, and `C:\NCSEXPER\DATEN\` for disassembled text files. When set, the specified folder is used instead. Please note that regardless of the setting this only applies to the path NCS Dummy will propose the first time you export a text file after application launch. If you decide to export to a different folder NCS Dummy will remember that folder for the next export operations.
- **Path to custom text file viewer and Text file viewer arguments** - When exporting or disassembling to text files, you can double-click the bold entries in the NCS Dummy log to quickly open the exported text files. By default NCS Dummy Microsoft Notepad to open these files. If you have a third party plain text editor you prefer, like Notepad++ or UltraEdit, point "Path to custom text file viewer" to the text editor's executable and (if necessary) specify the text editor's command line arguments in "Text file viewer arguments" (the default "%1" works for most applications). To get the most out of the exported files, the text editor must be set to a monospaced font like "Consolas", "Courier New", "Lucida Console",....
- **Repeat without confirmation** - Many file input fields in NCS Dummy have a reload feature. To reload the file either click "Browse..." and choose "Reload" from the button dropdown menu or press `CTRL+ENTER` or `F5` in the file input field. Some file input fields, like those on the "Disassembly and Checksums" tab, parse the input file and immediately ask for actions or saving the file. Reloading a file for such an input field suppresses the file open dialog but not later dialogs. When this option is enabled, NCS Dummy suppresses all dialogs while reloading and uses the last chosen actions and file names. The "Reload" action in the "Browse..." button dropdown menu is then replaced by a "Repeat" action to indicate the difference.
- **Check for newer version at startup *** - When enabled NCS Dummy checks for newer versions of the application and translations at startup. To reduce the amount of requests, the check is performed only once every three days. For those concerned with their privacy, rest assured that NCS Dummy does not submit data to the server. All the update check does is downloading text file `http://revtor.be/ncsdummy/ncsdummy-version.txt` which contains the latest version number and translations date. It is recommended to disable this option when NCS Dummy is used without Internet connection to speed up application startup.

6. NCS Expert

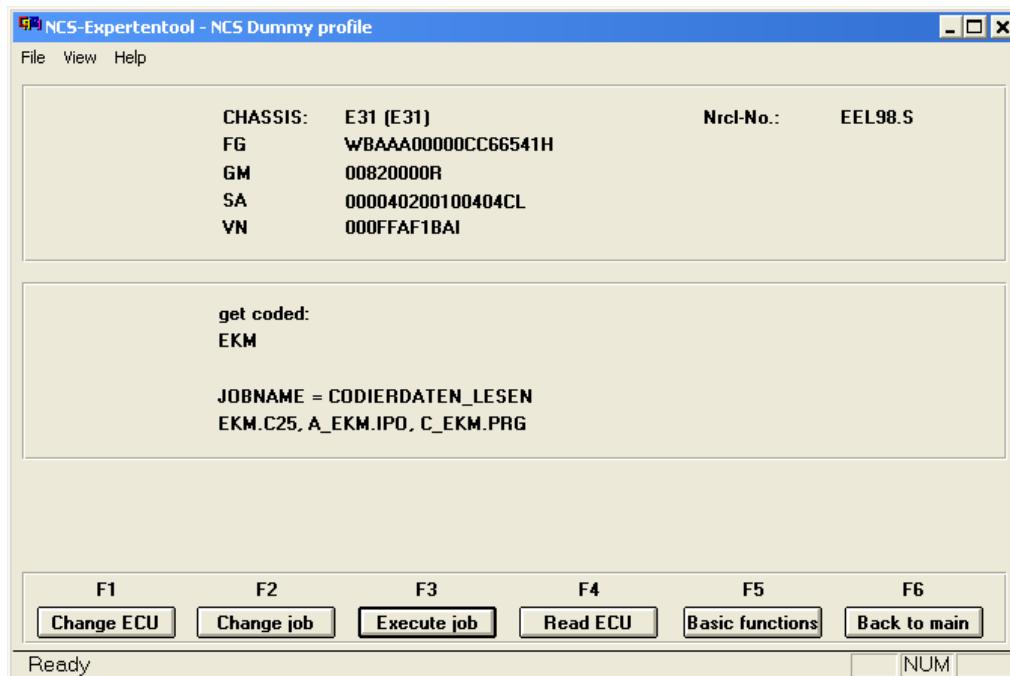
Before you can start reading or coding trace files with NCS Expert you need a profile that supports trace manipulation. The default profile that comes with NCS Expert, "Expertenmodus" (FZG2.PFL), does not support trace manipulation. If you do not already have a working profile, **download NCS Dummy profile for NCS Expert 3.*** (<http://revtor.be/ncsdummy/ncsexpert3-profile.zip>) or **NCS Expert 4.*** (<http://revtor.be/ncsdummy/ncsexpert4-profile.zip>) and extract NCSDUMMY3.PFL or NCSDUMMY4.PFL from the zip archive to the NCS Expert profiles folder C:\NCSEXPER\PFL\.

Navigate to the C:\NCSEXPER\WORK\ folder and (if not existing) create a new empty text file named FSW_PSW.MAN. This is the file used by NCS Expert to code through FSW/PSW manipulation. Even if you do not intend to code with NCS Expert, this file must be present! If NCS Dummy is installed on the same system you can also simply launch NCS Dummy to automatically create the trace manipulation file.

Unlike programming (flashing), coding does not take much time and no special precautions are required. There's no need for an external power supply while coding but make sure the batteries are charged and in good condition. Nevertheless it is recommended to hook up an external 13.8 V power supply for extended diagnosing and coding sessions. For old BMW models a simple battery charger with power supply mode functionality and a current output of 5 to 10 A or better should be sufficient, but newer cars have a much higher current draw. BMW recommends power supplies that can deliver over 40 A.

Now connect your diagnostic interface to the car and switch on the ignition (no need to start the engine). Test the connection with for example INPA. Don't forget to close INPA before using NCS Expert. The interface throws errors when multiple applications try to access it simultaneously.

Please note that FSW/PSW or nettodata coding does not change the vehicle central coding key (ZCS) or vehicle order options (FA). In other words; if the vehicle is ever recoded using the ZCS or FA all your changes will be lost. So if you are happy with the new coding, keep a backup of the FSW_PSW.MAN or NETTODAT.MAN trace manipulation file - just in case. It is also *strongly* recommended to keep a backup of the original unmodified FSW_PSW.TRC and NETTODAT.TRC trace files.



6.1. Reading FSW_PSW.TRC and NETTODAT.TRC trace files

The instructions below explain how to read the `FSW_PSW.TRC` and `NETTODAT.TRC` trace files from modules with NCS Expert. Despite some people's claims of how dangerous NCS Expert is, reading trace files from a module is perfectly safe as long as you keep your head at the job. NCS Expert's default setting is to code, which is definitely not wanted (at this point), but once the job type is changed to reading the application is harmless.

Please note that NCS Expert uses `FSW_PSW.TRC` as a temporary file for other purposes as well. In other words; most of the time the `FSW_PSW.TRC` trace file does not contain valid traces! Therefore it is important to follow the steps below carefully to end up with a proper trace file.

1. Launch NCS Expert. At this moment no profile is loaded, so the screen is quite empty.
2. From the "File" menu, choose "Load Profil".
3. In the "Choose profile" dialog box, select "NCS Dummy profile" and click "OK". Nothing much happens, but you'll notice some of the buttons on the bottom now have labels.
4. Click the "VIN/ZCS/FA" button (F1). The button labels change once again.
5. Click the "ZCS/FA f. ECU" button (F3).
6. In the "Choose CHASSIS" dialog box, select your chassis and click "OK".
7. In the "Choose ECU" dialog box, select one of the presented modules (for example "CAS", "EWS" or "EKM") and click "OK".
8. After a short moment, the top groupbox should now show information regarding your car.
9. Click the "Back" button (F6). The middle groupbox now shows the available modules and default job name. Depending on the used profile the default job is "SG_CODIEREN" (coding) which we definitely don't want, so pay attention to what buttons you press right now!
10. Click the "Process ECU" button (F4).
11. In the "Choose ECU" dialog box, select the desired module and click "OK". If you want to read all modules, you'll have to work module by module.
12. If "JOBNAME" in the middle groupbox already reads "CODIERDATEN_LESEN", skip the following steps and jump to 16.
13. Click the "Change job" button (F2).
14. In the "Choose job" dialog box, select "CODIERDATEN_LESEN" and click "OK". The default job action is now reading instead of coding. Note that NCS Expert will still say "get coded" in the middle groupbox and during the read operation it says it's coding... but it will not. "CODIERDATEN_LESEN" is a safe read-only operation.
15. Confirm "JOBNAME" in the middle groupbox says "CODIERDATEN_LESEN".
16. Click the "Execute job" button (F3). The process takes only seconds. As said, NCS Expert may indicate it's coding but it's actually only reading.
17. After the job completed, write down the `*.C??` module description file name displayed right below "JOBNAME = CODIERDATEN_LESEN" in the middle groupbox (for example `EKM.C25` for the E31 body electronics module). The `*.C??` file is the module name that will be used in NCS Dummy. It is important to do this after reading as the module description file shown before reading is the one NCS Expert expects and not necessarily the one that is actually present in the vehicle!

Steps 12 to 16 can be performed in a single step by clicking the "Read ECU" (F4) button instead but that always opens an annoying nettodata window... Newer versions of the "NCS Dummy profile" default to job "CODIERDATEN_LESEN" making steps 12 to 16 redundant.

Head over to the `C:\NCSEXPER\WORK\` folder and look for FSW/PSW trace file `FSW_PSW.TRC` and nettodata trace file `NETTODAT.TRC`. Do not perform any other actions in NCS Expert until the trace files are processed with NCS Dummy or saved to another location because NCS Expert uses `FSW_PSW.TRC` as temporary file for other actions and may overwrite the traces you just generated. If you rename the trace files using file name structure `CHASSIS-MODULE[-optionaltext].TRC`, for example `E31-EKM.C25-FSW_PSW.TRC` and `E31-EKM.C25-NETTODAT.TRC`, NCS Dummy will automatically set chassis and module when loading the trace files.

It is *strongly* recommended to keep a backup of the original unmodified trace files (before coding changes) for future reference.

6.2. Coding modules with FSW_PSW.MAN trace manipulation file

The most common way of coding modules with NCS Expert is through FSW/PSW manipulation. To do so, read a trace file from the module you wish to change the coding of, load it in NCS Dummy, make the desired changes in the graphical trace editor and export the `FSW_PSW.MAN` trace manipulation file. Then follow the instructions below to recode the module. If NCS Expert was still open after reading the trace file, start from step 12.

1. Launch NCS Expert. At this moment no profile is loaded, so the screen is quite empty.
2. From the "File" menu, choose "Load Profil".
3. In the "Choose profile" dialog box, select "NCS Dummy profile" and click "OK". Nothing much happens, but you'll notice some of the buttons on the bottom now have labels.
4. Click the "VIN/ZCS/FA" button (F1). The button labels change once again.
5. Click the "ZCS/FA f. ECU" button (F3).
6. In the "Choose CHASSIS" dialog box, select your chassis and click "OK".
7. In the "Choose ECU" dialog box, select one of the presented modules (for example "CAS", "EWS" or "EKM") and click "OK".
8. After a short moment, the top groupbox should now show information regarding your car.
9. Click the "Back" button (F6). The middle groupbox now shows the available modules and default job name.
10. Click the "Process ECU" button (F4).
11. In the "Choose ECU" dialog box, select the desired module and click "OK". Make sure the "FSW_PSW.MAN" manipulation file is for the same module.
12. If "JOBNAME" in the middle groupbox already reads "SG_CODIEREN", skip the following steps and jump to 16.
13. Click the "Change job" button (F2).
14. In the "Choose job" dialog box, select "SG_CODIEREN" and click "OK". The default job action is now coding.
15. Confirm "JOBNAME" in the middle groupbox says "SG_CODIEREN".
16. Click the "Execute job" button (F3). The process takes only seconds.

The module has been coded! Please note that some modules require the ignition to be switched off and on before the changes take effect. If you are happy with the result, keep a backup of `FSW_PSW.MAN` for future reference.

When switching between chassis it is recommended to empty (not delete) the `FSW_PSW.MAN` file after coding. Otherwise you may get strange errors when reading modules of another chassis. NCS Dummy can do this for you when closing the application and "Ask to empty trace manipulation file on exit" on the "Options" tab enabled.

6.3. Coding modules with NETTODAT.MAN trace manipulation file

Some functionality can only be achieved through nettodata coding. To do so, read a nettodata trace file from the module you wish to change the coding of, load it in NCS Dummy, make the desired changes in the graphical trace editor and export the `NETTODAT.MAN` trace manipulation file. Then follow the instructions below to recode the module. If NCS Expert was still open after reading the trace file, start from step 12.

1. Launch NCS Expert. At this moment no profile is loaded, so the screen is quite empty.
2. From the "File" menu, choose "Load Profil".

3. In the "Choose profile" dialog box, select "NCS Dummy profile" and click "OK". Nothing much happens, but you'll notice some of the buttons on the bottom now have labels.
4. Click the "VIN/ZCS/FA" button (F1). The button labels change once again.
5. Click the "ZCS/FA f. ECU" button (F3).
6. In the "Choose CHASSIS" dialog box, select your chassis and click "OK".
7. In the "Choose ECU" dialog box, select one of the presented modules (for example "CAS", "EWS" or "EKM") and click "OK".
8. After a short moment, the top groupbox should now show information regarding your car.
9. Click the "Back" button (F6). The middle groupbox now shows the available modules and default job name.
10. Click the "Process ECU" button (F4).
11. In the "Choose ECU" dialog box, select the desired module and click "OK". Make sure the NETTODAT.MAN manipulation file is for the same module.
12. Click the "Basic functions" button (F5).
13. In the "Choose basic function" dialog box, select "coapiCodeSgByNettoData" and click "OK".
14. In the following "coapiCodeSgByNettoData" dialog box, make sure the correct module is set for "CBD name" and enter NETTODAT.MAN for "Nettodataliste". Click "OK" to begin the nettodata coding job.

The module has been coded! Please note that some modules require the ignition to be switched off and on before the changes take effect. If you are happy with the result, keep a backup of NETTODAT.MAN for future reference.

If you get a COAPI-1063 error in NCS Expert, the module probably doesn't support nettodata coding. Click the "Change job" button (F2) and verify a job named "NETTODATEN_CODIEREN" is listed. If this job is not present, the module does not support nettodata coding.

6.4. Common NCS Expert errors

While working with NCS Expert you may run into some cryptic error messages. This chapter lists a few common errors and tries to explain how to avoid/fix them.

- **COAPI-1020: can not open file for reading** - Make sure the FSW_PSW.MAN manipulation file is present in the C:\NCSEXPER\WORK\ folder, even if you don't intend to code. If the file is seemingly present, Windows may be fooling you by hiding the real extension. You may be seeing FSW_PSW.MAN, but the file may actually be called FSW_PSW.MAN.txt which obviously does not work. Disable the Windows functionality to hide extensions for known file types and fix the file name.
- **COAPI-1063: CABD-parameter error** - This error occurs when you try to perform a job that is not supported by the module. The most common cause is trying to perform nettodata coding on a module that does not support it.
- **COAPI-1141: FSW faulty or COAPI-1142: PSW faulty** - These error messages are usually caused after the FSW_PSW.MAN file was forgotten to be emptied after coding. Empty (not delete) FSW_PSW.MAN and the error message should be gone.
- **COAPI-1143: error during processing of CBD-file** - If you modified NCS Expert DATEN files using the NCS Dummy disassembly tool, double-check your changes for errors. Otherwise check the FSW_PSW.MAN file for errors and make sure NCS Expert is using the correct module.
- **COAPI-2000: EDIABAS or SGBD error** - This one can point at many different errors. If "VIN is faulty" is shown as well, your interface is probably not configured properly or you are using the wrong interface for the wrong car. Otherwise the error may indicate outdated NCS Expert DATEN files. Update the files in C:\NCSEXPER\DATEN\, C:\NCSEXPER\SGDAT\ and C:\EDIABAS\ECU\ to the latest versions.
- **COAPI-2020: no answer from ECU** - You are trying to communicate with a module that is not installed or dead.

7. Changelog

What's new since the last release?

0.6.0.1 (2013-08-07)

- Fixed "MaxDropDownItems" crash at very high display resolutions.
- Fixed broken shortcut key CTRL+C to copy a single line from the log.
- Added the more familiar shortcut key F5 as an alternative to CTRL+ENTER to reload dropdowns and path textboxes.
- Shortcut key F3 to find the next search item now also works when the focus is not on the graphical trace editor.
- Searching in the graphical trace editor now scrolls the entire function into view (if possible) instead of only the found item.
- Find window opens automatically after successfully filtering modules by keyword or order option.
- Small number of new and corrected translations.

0.6.0.0 (2013-02-03)

- Completely redesigned user interface. All user interface changes would make a list larger than the combined change logs of all previous versions. Listed below are just the highlights.
 - Resizable and maximizable application window. Window size and location are saved between sessions.
 - Moved the graphical trace editor to the "Modules and Traces" tab and removed the "Trace Editor" tab.
 - The graphical trace editor is now also used to display module contents or changes between trace files.
 - The graphical trace editor is no longer sorted alphabetically but exactly as defined by the module description files. Together with the addition of groups this should make the trace editor contents much more logical.
 - The graphical trace editor now shows the complete module contents for nettdata trace files or all functions in the module for FSW/PSW trace files. Items that do not occur in the loaded trace file are shown in a disabled state, but can be enabled and included in exported trace files.
 - Many "Browse..." buttons have now dropdown menus with shortcuts to useful functions.
 - "Path to NCS Expert DATEN folder" moved to "Options" tab.
 - Second trace file input field hidden by default to save space for the graphical trace editor. To compare trace files either use the trace file 1 "Browse..." button dropdown menu or unhide the second trace file input field on the "Options" tab.
 - Checkboxes to export translations, control unit data and order options moved to the dropdown menus of the "Chassis Functions", "Module Functions" and "Trace Differences" buttons. The settings are saved individually for each button.
 - Usage of order options is now enabled by default. This has a severe impact on the loading times of large chassis (like E60 and E89). If you think this is unacceptable disable "Load order options" on the "Options" tab but keep in mind you miss out on a lot of information.
 - Removed functionality to export function, parameter and option keyword lists. For those who used this to find the highest keyword identifier, the disassembly tool will now automatically add a commented entry with the next highest identifier at the end of disassembled keyword lists.
 - NCS Dummy now follows the system font and has improved DPI awareness. Since Windows Vista/2008/7/8 use a slightly larger font than Windows XP/2003 the application is scaled larger on these operating systems. This has a negative effect on the available space for the

graphical trace editor. For users of small displays command line argument `/smalldisplay` was added which decreases the font and log size to maximize the available space for the graphical trace editor.

- The registry path where NCS Dummy stores its settings has been changed to `HKEY_CURRENT_USER\Software\revtor\NCS Dummy\`.
- Remember last loaded chassis between sessions (can be disabled on the "Options" tab).
- Filter modules by keyword or order option.
- Filter modules by FSW/PSW or nettodata trace file.
- Add parameters to functions from within the graphical trace editor and update the actual NCS Expert data files with the changes. This should greatly reduce the need for manual disassembly and assembly. Backups are created on the first change and can be restored from within NCS Dummy.
- Multiword functions are now linked. All changes to a multiword function are duplicated in its counterpart.
- Ability to search backwards in the graphical trace editor by holding the SHIFT key while searching.
- Ask to empty `FSW_PSW.MAN` trace manipulation file on application exit (can be disabled on the "Options" tab).
- 10000+ new translations.
- Small changes to the output of disassembled DATEN files (formerly called Ediabas files). Files disassembled with previous versions are not compatible with this version.
- Checksum updater for ZST files.
- Checksum updater for WinKFP/NFS files.

0.5.0.5 (2011-11-07)

- Fixed bug in disassembly tool. The "AUFTRAGSAUSDRUCK" field in `*CVT.000`, `*SGET.000` and `*SGVT.000` files was not processed correctly.
- Drag and drop support for file path textboxes.

0.5.0.4 (2011-11-02)

- Fixed bug in trace editor tree view. In some cases the tree view failed to render properly on Windows Vista and Windows 7 systems.

0.5.0.3 (2011-10-26)

- Fixed bug in trace editor nettodata export. In some cases the exported nettodata trace file contained incorrect addresses.

0.5.0.2 (2011-09-01)

- Default text export folder. When set, NCS Dummy will use the specified folder as default export folder for text files instead of the `WORK` or `DATEN` folder.
- Find by parameter keyword in the trace editor now works.
- DPI awareness. NCS Dummy will now scale correctly on displays with a dot pitch different from the default 96 dpi.
- Improved usability on very low resolution displays. The application will now calculate the available screen space and if necessary resize and reposition itself accordingly.

0.5.0.1 (2011-08-27)

- Find functionality for the trace editor. You can search on keyword, hexadecimal data, address or order option in the currently loaded trace file. Use shortcut key CTRL+F to open the find window.
- Small changes to the disassembly output. Please note that even though the changes are subtle, files disassembled with previous versions are not compatible with this version.

0.5.0.0 (2011-06-14)

- Use of background threads. Long operations are now performed on a background thread which keeps the user interface responsive. Progress is indicated by a progress bar in the header of the application.
- New parser for NCS Expert data files. It turned out the new parser introduced in version 0.4.0.0 did have some issues left.
- Support for all types of nettodata trace files (`NETTODAT.TRC`). The previous version only supported nettodata trace files with single byte data values.
- Support for the partially corrupt NCS Expert data files `KMBI_E38.C12` and `ARS.C0E`.
- Strict mode for the parsing of keyword trace files (`FSW_PSW.TRC`) in which no unresolved keywords are allowed. This feature can be turned off in the options.
- Cross references to the order options. When enabled, NCS Dummy will now include the order options for each parameter (if available). This should be a tremendous help for the otherwise meaningless "wert_01", "wert_02",... parameters. The cross referencing is CPU intensive and may cause slow downs on slower machines for the E60, E65 and especially the E89 chassis.
- Graphical trace editor. When a keyword trace file (`FSW_PSW.TRC`) or nettodata trace file (`NETTODAT.TRC`) is loaded, a new tab "Trace Editor" appears. In the trace editor you can view and modify the trace file. When starting from a nettodata trace file you can even add custom parameters. Now you no longer have to do all the hexadecimal mask calculations by yourself. NCS Dummy does everything for you.
- Disassembly and assembly of NCS Expert data files. Ever wondered what is in the NCS Expert data files or perhaps you even wanted to modify something? Now you can. The functionality can be compared to the "data2txt" application by "russiane39". Please note that while the syntax is similar, it is not compatible.
- NCS Dummy now stores its preferences in the Windows Registry (`HKEY_CURRENT_USER\Software\revtor software\NCS Dummy\`).
- Ability to set a custom text editor to open the exported files into.
- Automatic check for newer versions of NCS Dummy and/or the translations at startup. For those concerned with their privacy, the feature can be disabled in the options, but all it does is downloading text file <http://revtor.be/ncsdummy/ncsdummy-version.txt> which contains the latest version number and translations date. No data is sent from your computer to the server.
- Check for the correct Microsoft .NET Framework version at startup. This was already implemented in 0.2.1.1 but somehow got lost in 0.4.0.0.
- Many, many other small improvements...

0.4.0.0 (2010-11-05)

- New and much improved parser for NCS Expert data files. The old parser used masks to locate and identify data, whereas the new parser truly interprets the data file format. No more guessing.
- Ability to display control unit data: This feature may be of use only to a small number of users, but it's what makes NCS Dummy extremely powerful. NCS Dummy can tell you what data to expect at a certain address location in the module's coding memory and to a limited extent also what it means. This is a great aid if you intend to write custom data to a module to accomplish features impossible with the classic `FSW_PSW.MAN` trace manipulation.
- Support for nettodata trace files (`NETTODAT.TRC`): Because the new version is able to handle control unit data, it can also parse nettodata trace files. Nettodata trace files often contain more information than the dumbed-down keyword trace files (`FSW_PSW.TRC`). Nettodata trace files are a more reliable representation of the control unit's coding. Keywords may change between NCS Expert DATEN versions, but the binary data always remains the same.
- The new version maps nettodata and keyword trace files against the NCS Expert module file. The old versions used the module file only as reference and ignored them for trace differences. That makes the new version less tolerant for module version mismatches. Make sure you select the same chassis

and module as used in NCS Expert to generate the trace file. You can still compare trace files from different modules, but expect a lot of unresolved functions and parameters.

- Automatic detection of chassis and module from the keyword or nettodata trace file name. If you rename your trace files to a CHASSIS-MODULE-FILENAME.TRC structure (for example E31-EKM.C25-FSW_PSW.TRC), NCS Dummy will automatically set the chassis and module (chassis E31 and module EKM.C25 in the example). This should save time when working with many trace files.
- Full export functionality for the second trace file. In previous versions the second trace file could only be used for the differences.

0.2.1.1 (2010-09-01)

- NCS Dummy no longer uses a hardcoded list of supported chassis, but scans the NCS Expert DATEN folder for chassis instead.
- Check for the correct Microsoft .NET Framework version at startup.

0.2.1.0 (2010-07-07)

- NCS Dummy now supports all known chassis: E31, E32, E34, E36, E38, E39, E46, E52, E53, E60, E65, E70, E83, E85, E89, K24, R50, R56, RR1 and derived chassis.
- Support for translations of function and parameter keywords in the exported files.

0.2.0.11 (2010-07-02)

- Added support for chassis E60.

0.2.0.10 (2010-06-30)

- Added support for chassis E89.
- After a successful export you can now double-click on the log line to open the exported file immediately in the system's default application for text files.

0.2.0.9 (2010-06-28)

- Added support for chassis E85.
- Improved NCS Expert data file parser.

0.2.0.8 (2010-06-27)

- First public release.